ON-LINE SIGNATURE RECOGNITION USING A GLOBAL FEATURES FUSION APPROACH

Eugen LUPU[1] Simina EMERICH[1] Francois Beaufort[2]
[1]Technical University of Cluj-Napoca, Communications Dept.
26-28 Baritiu str.,Cluj-Napoca, Romania, Phone: +40-264-401266, Eugen.Lupu@com.utcluj.ro
[2]Ecole Supérieure d'Electronique de l'Ouest, Angers, France

Abstract: The paper presents a feature-based approach regarding on-line signature recognition, by employing the TESPAR - DZ matrices extracted from every signature. These matrices describe the different signature features shapes, such as x, y - coordinates, x, y - velocity, pressure etc. For better results, the fusion of different features was studied. Three signature databases have been employed in our experiments: DB1, DB2, our own signature databases and the Task2 database available for research purposes from the SVC2004 contest. In our work, two types of experiments were made: verification and identification (surveillance). These results are very promising taking into account the simplicity of template generation and the fact that we employed only five features in our experiments.

Key words: *on-line signature, verification, identification, TESPAR, fusion, forgery, SVM.*

I. INTRODUCTION

Biometric recognition is adverted to the automatic identification of a person based on some specific vectors, derived from the biological characteristics. Due to the fact that biometric verifiers can't be easily counterfeit, borrowed or unsuited keep, they are considered to be more secure for the person's verification than the traditional methods.

Usually, a classification of biometric features is made in: physiological features (fingerprint, face shape, iris, retina etc.) and behavioral features (voice, gait, writing style etc.). In practice, all biometric verifiers may be considered combinations of physiological and behavioral characteristics due to the interaction mode between the user and the system, which puts its mark over the characteristic.

People are used to signatures as a means of transaction-related identity verification, and most would see nothing unusual in extending this to encompass biometrics.

Signature is a trait that characterizes a single individual so it differs from other biometric systems. Signature verification system analyzes the way a user signs his/her name. These systems are reasonably accurate in operation and obviously lend themselves to applications where a signature is an accepted identifier.

There are two types of signature biometric systems according to the input signature information: *on-line* and *off-line* categories [1].

On-line or dynamic signatures are written with an electronically instrumented device and the dynamic information (pen tip location through time) is usually available at high resolution, even when the pen is not in contact with the paper. On-line signature methods take consideration the time functions of the dynamic signing process, such as: velocity, velocity, trajectories or pressure versus time. These features are as important as the signature's static shape

Off-line or **static** signatures use the static image of the signature, which is scanned from paper documents, where it was written in a conventional way.

The reason why On-line signature verification is more popular and more reliable than the Off-line signature verification is that a forger can mimic the shape of the signature but it is very hard for him to copy the shape as well as its dynamic features [6].

Signature verification from the point of view of biometric features presents some advantages, such as:

- Non invasive
- User-friendly; enrollment (training) is intuitive and fast
- Well accepted socially and legally
- Signature verification generally has a fast response and low storage requirements
- High compression rates do not affect shape of the signature
- Acquisition hardware:
  - Off-line: ubiquitous (pen and paper)
  - On-line: already integrated; cheap devices (Digitizing Tablet)
- The Signature, if compromised, can be changed
- The Signature is a man-made biometric where forgery has been studied extensively.

Like other biometrics, signature presents also some disadvantages, such as:

Manuscript received August 24, 2009; revised October 14, 2009

• the signature has high user intra-variability
• Large temporal variation.
• possibility of forgeries
• higher error rates than other traits
• affected by the emotional and physical state of the user
• Some people have palsies, while others do not have enough fine motor coordination to write consistently [1] [2].

II. ON THE ON-LINE SIGNATURE RECOGNITION SYSTEMS

A typical architecture of an on-line signature recognition system is depicted in figure 1.

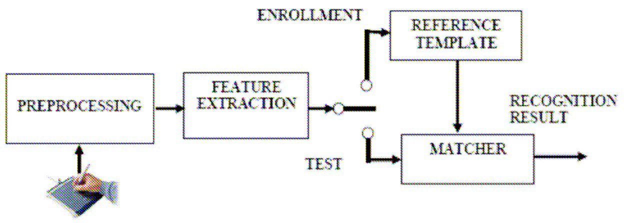

Figure 1. Signature recognition system architecture

The most important system modules, the main techniques employed in the study and the related issues will be summarized in the next sections.

Data acquisition and preprocessing

Various kinds of devices are used to capture signature dynamics, such as traditional tablets, touch screens or special purpose devices. Special pens are able to capture movements in 3 dimensions.

Generally, pressure-sensitive tablets are used to capture the image and dynamic properties of the signature, as this representation is more unique and more difficult to forge.
Tablets employed to capture 2D coordinates and pressure have two significant disadvantages. Usually, the resulting digitalized signature looks different from the traditional user signature; also, sometimes while signing the user does not see what has been written so far. This is a considerable drawback for many users.

The on-line acquisition devices usually operate at 100-200 samples/sec, based on the fact that biomechanical sequences are always at less than 20-30Hz frequency. So the Nyquist sampling criterion is observed and this sampling frequency leads to a sharp discrete-time signature representation. Usual tablets provide at each sample point: the x, y coordinates of the signature's trajectory, pressure, pen azimuth, pen altitude versus time.

Before feature extraction, some preprocessing steps are generally required. Common preprocessing techniques employed in signal enhancement are:
- Resampling in order to reduce number of points and in order to obtain equidistant points capturing the shape;
- Gaussian filtering to smooth signature to remove signal noise;
- Critical points (trajectory change, start/end points) extraction to avoid information loss; these positions will be retained before resampling.

Feature extraction

Many approaches have been employed in order to extract the representative information from the on-line signature data.

On-line signature verification methods can be broadly divided into two groups: function-based methods and feature-based methods.

Function-based methods use time sequences to describe the local properties of the signature. With these methods, the signature is stored as a discrete function to be compared to a pattern from the same user, previously computed during an enrolment stage. Such methods simplify data acquisition but comparison can become a hard task.
The local features may be:
–x, y coordinates
–Velocity (v)
–Acceleration (A)
–Trajectory angle (Θ)
–Azimuth
–Elevation
–First derivative of feature (Δ)
–Second derivative of feature (ΔΔ).

One local property based system, mentioned in [5], relies on three pseudo-distance measures (shape, motion and writing pressure) derived from coordinate and writing pressure functions through the application of a technique known as Dynamic Time Warping (DTW). It is reported to have over 90% success rate. Another approach is the use of Fast Fourier Transform as an alternative to time warping. It is suggested that working in the frequency domain would eliminate the need to worry about temporal misalignments between the functions to be compared. It is concluded that the FFT can be useful as a method for the selection of features for signature verification.

Feature-based methods use global features extracted from signature identified trajectories. Preparing data for comparison may prove to require extensive efforts. Yet, the actual comparison process is highly effective.

The global features may be more than 150 [2]. Some of them may be the following:
–Signature length, height, weight
–Average velocity
–Maximum velocity
–Minimum velocity
–Maximum pressure
–Average pressure
–Average velocity x
–Average velocity y
–Total signature time
–Total pen-down time
–Total pen-up time.

Users Enrollment
Depending on the envisaged matching strategy, the enrollment may be: reference-based and model-based.

In reference-based enrollment, the features extracted from the training set are stored as a set of patterns, each one corresponding to one training signature. For the matching process, the features extracted from the test signature are compared with all the reference templates and the resulting matching scores are combined with a score-level fusion technique. This enrollment mode is more appropriate when the training set is small. It is generally accepted that the 5 signatures for the training set in on-line signature verification represents the best choice [3].

In model-based enrollmen,t the training set of a user is employed to provide a statistical model which describes the signing manner of the given user.

During the enrollment, an important problem consists of the variation of signatures in time. To alleviate this problem, one can employ signatures from different sessions or use model adaptation in another approach [3].

Similarity computation
Before the matching level, usually an alignment between the test signature and the user model or template is made.

For the reference-based enrollment, the alignment is usually performed before the feature extraction based on the signature shape is made. When the model-based enrolment is employed, the pre-alignment in applying a common system reference is used (e.g. trajectories position regarding to the initial point or to the center of mass, scaling to a fixed frame size etc) [2]. If no alignment is employed, this is incorporated in the matching procedure.

In feature-based systems with reference-based enrollment, the matching scores are usually obtained by computing distances between feature vectors of test and template signatures or by training a classifier. The most employed distances are: Euclidian and weighted Euclidian distance and Mahalanobis distance. In feature-based systems employing model-based enrollment, statistical models using non-parametric density estimation have been used.

Function-based approaches can be divided, depending on the matching strategy, into local and regional.

In local approache,s the elastic DTW (Dynamic Time Warping) procedure is used to match the time functions of different signatures.

For regional methods, the time functions are transformed in a vector string, every one describing the regional properties of a signature segment. Hidden Markov Models (HMM) is one of the most employed approaches for the regional methods.

Score normalization
The matching scores provided by comparing the test signature with the template/model signature are generally normalized to a common range such as [0, 1], employing different mapping functions before comparing them to a decision threshold. The score normalization can be user-dependent, like in other system modules, in order to prevent misalignments which can be compensated by user-dependent thresholds.

III. TESPAR CODING METHOD BACKGROUND

Feature extraction involves simplifying the amount of resources needed to describe a large set of data. Analysis with a large number of variables generally requires a large amount of memory and computation power. Therefore, the main purpose in the feature extraction process is to generate features that exhibit high information packing properties.

An overview of TESPAR coding method may be found in references [7],[8]. This section makes a short presentation of the TESPAR method and brings new information on the TESPAR DZ version.

The key of the TESPAR method is that precise description of the waveform does not require polynomials, but uses an approximate model based on the zeros theory. The waveform is divided in periods determined by successive passes through zero of the signal, thus maintaining the time information combined with a simple approximation of the waveform in-between two successive passes through zero.

Real zeros in the time domain are identical to the location of real zeros in the frequency domain. Complex zeros are conjugated and associated to noise: minima, maxima, turning points, that appear on the waveform in-between two successive passes through zero. Thus, analyzing the waveform in-between two successive passes through zero we may obtain a sufficient number of subsets of zeros to approximate the studied waveform.

The simplest implementation of a TESPAR coder uses two descriptors associated to each epoch (waveform segment situated between two successive passes through zero):
- ***duration (D)*** between two successive zero crossing of the signal (in samples)
- ***shape (S)*** of the signal between two successive zero crossings.

In the simple TESPAR model, not all the zeros may be determined out of the waveform; therefore, *the model is limited to those zeros that can be figured out directly from the waveform.*

The limited band of the signals imposes significant restrictions on the minimum and maximum period of the epochs and on the number of important characteristics of the waveform associated to each epoch (minims, maximums, points of inflexion, etc.). The longest epoch may have a period up to approximate half the period of the lowest frequency accepted by the limited band signal, while the shortest period is half the period of the highest admitted frequency. It seems obvious that short epochs cannot have a large number of minima, as this would imply crossing the limited band of the signal.

A first approximation considers each epoch characterized *by its duration **D** and the number of minima (shape) **S**.*

In order to obtain an even better compression and to avoid a disturbing part of the variability of the epochs quite alike, the vector sampling process is used obtain a TESPAR alphabet. Thus we obtain a table of symbols that is further used in the coding process to associate a symbol to each epoch described by the two descriptors D and S [8].

The table of symbols is usually built so that we obtain the best approximation of the most probable epochs – this means its goal is to minimize the general distortion generated by the quantization process. First, we need to determine a statistical distribution of the epochs and then draw the table of symbols before using the TESPAR coding.

It is common to use a recording of the waveform able to provide statistical information on the epoch in the domain of interest of the application.

Practical experience has proved that the number of symbols that is large enough to successfully approximate the signal (in classification applications) using this method is 30. The TESPAR coder issue is a string of symbols computed based on the descriptors D and S associated to the epochs, fig.2 [8]. This simple series of symbols may be converted to fixed length structures that are easy to be used.

One can use a one-dimensional matrix (vector) that counts the number of appearances of alphabet symbols in an

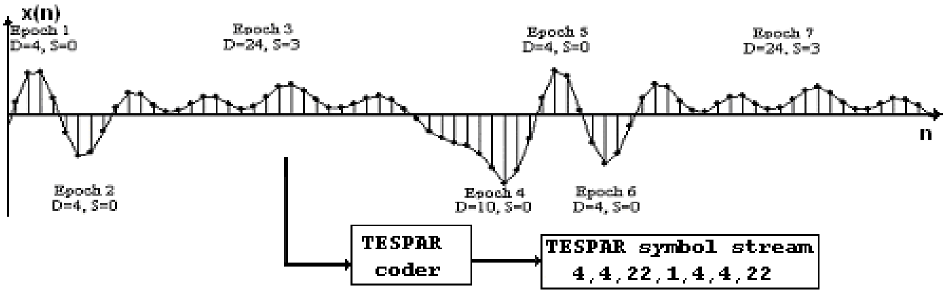

Figure 2. TESPAR coding process

utterance and a histogram is obtained, the TESPAR-S matrix.

Another standard bidimensional TESPAR matrix is the TESPAR-A matrix. It is a matrix that counts how many times two symbols, not necessarily adjacent, are situated at a certain distance from each other. In this case another parameter called *lag* is required to provide information on the short-term time evolution of the analyzed waveform if its value is less than 10 or on the long-term evolution if its value is higher than 10. This bidimensional matrix assures a greater discriminatory power.

The discriminatory power may be improved by using a matrix with three dimensions. The main advantage of processing signals using the TESPAR method over traditional methods based on frequency descriptors is that *TESPAR matrices are fixed length structures.*

There are two main methods of classifying using TESPAR: classifying using archetypes and classifying with neuronal networks.

This paper employs a version of this method based on TESPAR DZ matrices. In this case three descriptors will be employed to describe every epoch: D, S and A (Amplitude) which designates the maximum value in the samples of an epoch. Employing TESPAR DZ coding procedure, pairs of epochs are compared; whereby each type of descriptor, from each epoch pair, is compared and a symbol is produced indicating the differences between the individual D, S and A, features of the two epochs being compared [9].

The epochs descriptors comparation may be done for different lags. For example, with a lag of 1, comparisons will be made between epoch K and epoch K-1, while for a lag of 2, comparisons are made between epoch K and epoch K-2, and so on. Figure 3 presents the flowchart of the entire coding procedure or symbol assignation.

For each individual epoch pair comparison, a three-stage comparison vector is generated for each epoch descriptor. Thus for a lag = 1, when comparing D, S and A, for epochs E1 and E2 successively, the following value results:

For D2 versus D1: if D2=D1, the resulted value is 0
if D2<D1, the resulted value is -1
if D2>D1, the resulted value is +1.

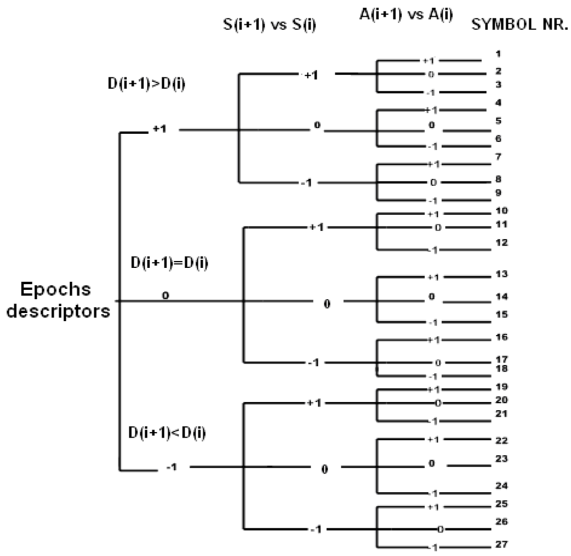

Figure 3. TESPAR DZ symbols assignation

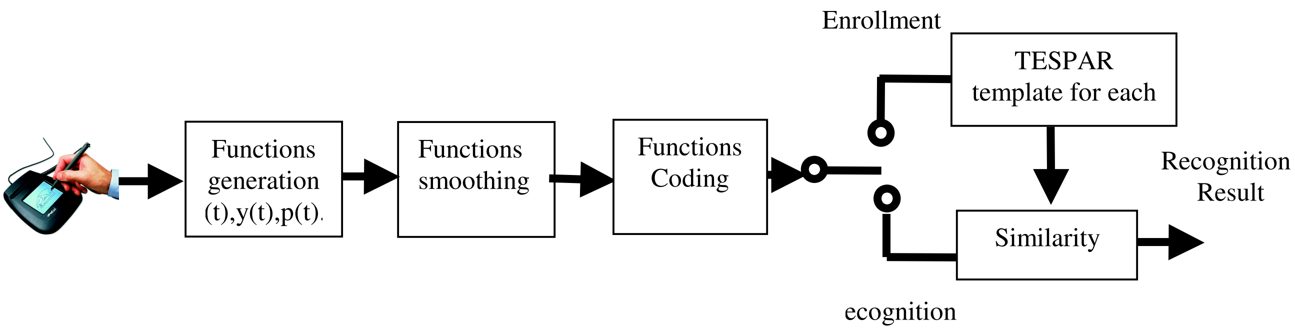

Figure 5. The system architecture

By using this algorithm, from any paired descriptors comparison of D, S and A (belonging to each epoch), one of the 27 possible difference options may be derived, indicative of the nature of difference between the pair of epochs under investigation. These symbols were arbitrarily assigned to the numbers 1 to 27 of the 27-symbol DZ TESPAR alphabet.

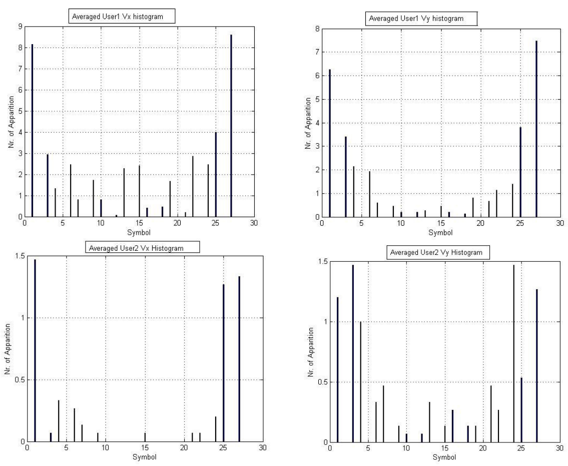

Figure 4. TESPAR DZ matrices template for two users (features Vx and Vy)

IV. THE EXPERIMENTAL SETUP APPROACH

The system block diagram of our approach is depicted in fig.5. The processing steps follow the schedule presented in this figure. First, the data rows from the tablet (or from the database) are used to generate the time functions that we decided upon, so as to provide the best results in our approach. These functions are often very jagged and in order to remove this noise they are smoothed by filtering, employing a median filter. After some experiments, we decide to employ the following functions: y(t), v_x(t), v_y(t), p(t) to generate the templates. The next figure shows the variation of several parameters, considering two different users.

Then, the selected functions are coded by the TESPAR DZ coder and provide the TESPAR matrices or histograms. So we may consider these matrices to be the features which characterize the time functions of the signature.

To generate the models for every user enrolled in the experiment, these matrices are averaged for the users training set or are employed for training a classifier.

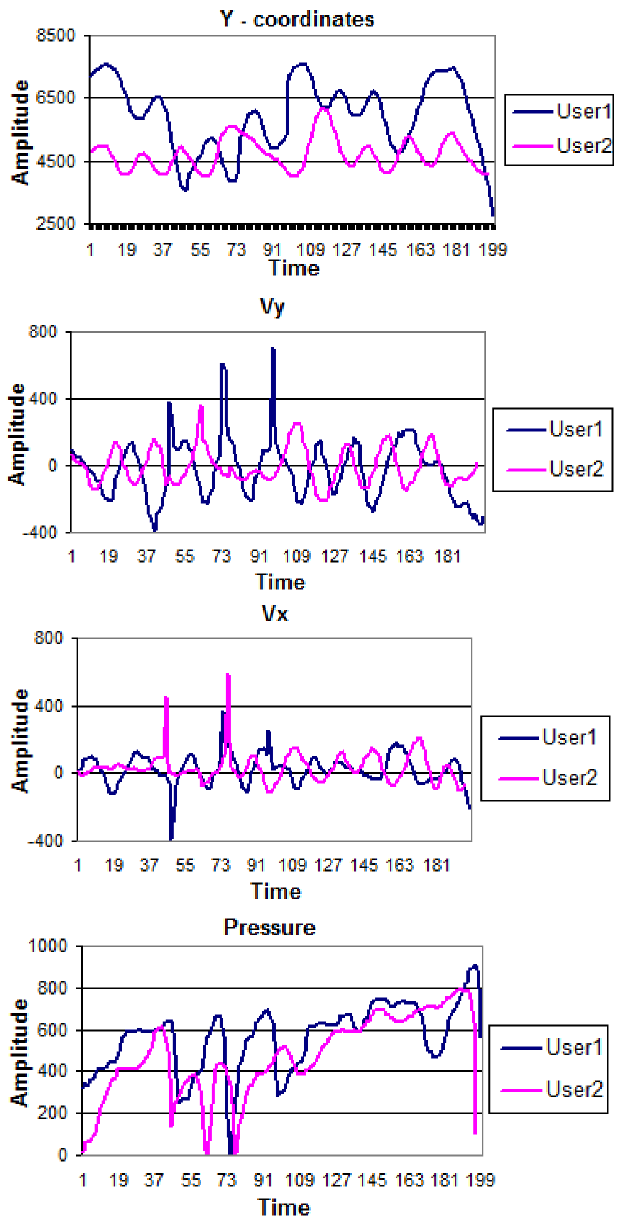

Figure 6. The functions y (t), v_x (t), v_y (t), p (t) timing for two different users

Weka LibSupport VectorMachine software toolbox

To perform learning and classification tasks for signature verification and identification, WLSVM software toolbox has been employed (Weka LibSVM - Integrating LibSVM into Weka Environment). Weka and LibSVM represent valuable software tools in the design of Support Vector Machine classifiers. One must consider both advrantages and disadvantages of using each tool. LibSVM supports various SVM methods and works considerably faster than Weka. The latter has a friendly GUI and computes important statistics, such as recall, precision, confusion matrix, F-measure or ROC scores. WLSVM is a combination of these two tools and may be seen as a form of implemented LibSVM working in Weka environment [11].

SVMs are well known in the pattern recognition research area due to their generalization capabilities achieved by structural risk minimization oriented training. WLSVM offers the option to change the kernel for the SVM classifier. In our experiments the RBF kernel outperformed the polynomial one.

Two parameters are to be considered with RBF, their being C, the penalty parameter of the error term, and γ, the gamma – kernel parameter. It is hard to decide which of these two parameters will better meet the needs for one situation. Therefore, parameter search is performed for model selection. The standard method to determine the best solution is by grid search. In our experiments the simple grid-search approach was used: parameters are varied with a fixed step-size through a wide range of values and the performance of every combination is measured [12].

Signature Databases

Each signature from the used databases (DB1, DB2 and Task2-SVC2004) is characterized by *X coordinate*, *Y coordinate*, *time stamp* and *pressure*. Additional information of pen status, azimuth and altitude from SVC 2004 database was ignored in our experiments.

We built our own signature database **DB1 (only genuine signatures)** which contains signatures from 30 individuals (women and men). The signatures were collected with a digitizing tablet (GENIUS G-Pen 4500), without any constraints on how to sign. The signers were students or staff members of Technical University of Cluj-Napoca, with ages varying between 20 and 50. Each signature contributor supplied 15 samples in three different sessions. There was an approximately two weeks time period between the sessions. Signers were suggested to practice before the actual data collection started. A signature is represented by a sequence of points and each point is described by several features: *X-coordinate, Y-coordinate, time stamp and pressure.* Using these time functions, we were able to extract other functions such as velocity and acceleration.

Another signature database **DB2 (genuine signatures + skilled forgeries)** involving 10 sets of signature data was also created. Each set contained 20 genuine signatures from each user and 20 skilled forgeries. The skilled forgeries for each data contributor were provided by four other signers in the following way. Using a software viewer, the genuine signature could be seen. Signers were advised to practice the skilled forgeries several times until the data collection.

A publicly available handwritten signature database was also employed in this research. This corpus provided by the First International Signature Verification Competition **(Task2-SVC2004)** consists of 40 sets with 40 signatures for each set (http://www.cse.ust.hk/svc2004/download.html).

The enrolled users were advised to write naturally on the digitizing tablet (WACOM Intuos tablet) as if they were enrolling themselves to a real signature verification system.

Contributors were provided the option of not accepting a signature instance if they were not satisfied with it. Each set contains 20 genuine signatures from each signer and 20 skilled forgeries.

In each text file, the signature was simply represented as a sequence of points. The first line stored a single integer which is the total number of points in the signature. Each of the subsequent lines corresponded to one point characterized by:

X-coordinate - scaled cursor position along the x-axis
Y-coordinate - scaled cursor position along the y-axis
Time stamp - system time at which the event was posted
Button status - current button status (0 for pen-up and 1 for pen-down)
Azimuth - clockwise rotation of cursor about the z-axis
Altitude - angle upward toward the positive z-axis
Pressure - adjusted state of the normal pressure [10].

V. EXPERIMENTS AND RESULTS

Our first purpose was to find a suitable combination of features in order to improve a classifier's ability to recognize forgeries, to tolerate intrapersonal variances and also to identify signatures.

It is commonly accepted that the signature dynamic information such as the velocity, acceleration or pressure is difficult to forge and thus can be used to distinguish skilled forgeries. Therefore the first-order difference functions *Vx* (velocity of the *X*-coordinate) and *Vy* (velocity of *Y* - coordinate) were computed by subtracting neighbouring points. From the velocity sequence, acceleration (*Ax* and *Ay*) was further derived. The lengths of used features were variable. Further a new algorithm for on-line handwriting signature verification and identification is proposed.

Experiments for feature selection

In order to evaluate the effectiveness of the system several experiments were carried out. The first series of experiments shows a classification process of the DB1 signatures as belonging to a particular individual. Different feature vectors were used. Each of them has a 27 coefficients dimension and is obtained from a TESPAR DZ matrix generated from the mentioned parameters.

The most suitable parameters combination for the WLSVM was: C=100; γ=0.001 for the RBF kernel and C=100; d=1 for the POLY Kernel. The classification rates issued by experiments are presented in the next figure.

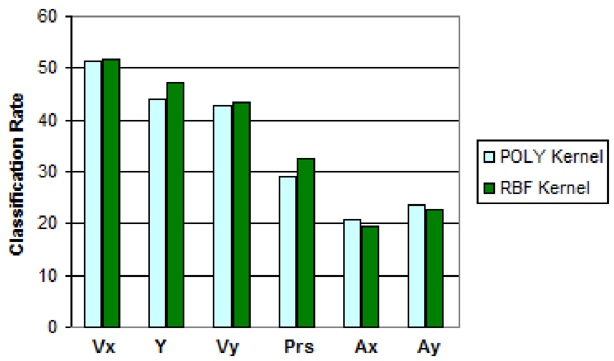

Figure 7. Signature identification rates issued for different parameters

These experiments show that the features like *Y* - coordinates, the writing velocities *Vx, Vy* and the pressure *Prs* are among the most consistent. From the confusion matrix, it was also observed that the same parameters have different relevance among subjects.

The results reveal also that the system based on RBF Kernel gave better classification rates than the one based on Poly Kernel.

Signature identification experiments using features fusion

The system's goal is to obtain a small feature vector and a good classification rate. In this order, it will be shown that by the fusion of above most consistent parameters an improvement of the recognition accuracy can be achieved. In addition, the number of epochs of each feature is considered.

Since RBF Kernel seems to be the most suitable classifier, we used it for further experiments. The classification rates that were established for DB1 signatures identification are summarized in the next table.

Feature vector						Classification Rate
Parameters					Length	
Vx	Y	Vy	Prs	Ep		
✓	✓				2*27	69.77%
✓	✓			✓	2*27+2	74.66%
✓	✓	✓			3*27	73.77%
✓	✓	✓		✓	3*27+3	78%
✓	✓		✓		3*27	76.44%
✓	✓		✓	✓	3*27+3	80.22%
✓	✓	✓	✓		4*27	78.44%
✓	✓	✓	✓	✓	4*27+4	**84%**

Table 1. Identification rates for ***DB1*** *using features fusion*

There are inherent variations in the signature patterns written by the same person. The variations can occur in the shape or in the relative positions of the characteristic parameters. Therefore, the new feature extraction technique has been extensively explored.

The results obtained indicate that the proposed system is able to identify signatures with great accuracy (84%), by using a 112 coefficients feature vector, obtained by the fusion of the coefficients provided by the following features: *V_x, Y, V_y, Pressure, number of Epochs.*

Signature verification experiments

Unlike physiological biometrics, the use of skilled forgeries is very important to behavioural biometrics such as handwritten signature. Consequently, the second part of this research involves verification. Verification is the decision about whether the signature is genuine or forgery. To perform learning and classification tasks for genuine and skilled forgeries signatures, the **Task2-SVC2004** database was employed.

Further, we report the best results of our approach for the on-line signature verification task. The technique was based on RBF Kernel of SVM classifier and the feature vector dimension was of 4*27+4 coefficients. Another remark is that the number of epochs seems to be, in this case, very suitable.

	Number of users	Verification Rate [%]	Detailed Accuracy	
			HTER	RMSE
	2	100	0	0
	4	97.5	0.025	0.158
	5	95	0.05	0.223
	8	92.5	0.075	0.273
	8	90	0.1	0.316
	9	87.5	0.125	0.353
	4	85	0.15	0.387
Weighed Average	**40**	**91.31%**	**0.137**	**0.279**

Table 2. System verification rates for the ***Task2-SVC2004*** *Database*

A two-by-two confusion matrix (also called a contingency table) was constructed for every user. This matrix represents the dispositions of the set of used instances (20 genuine signatures and 20 skilled forgeries) and forms the basis for many common metrics. The Half Total Error Rate, HTER is defined as [13].

$$HTER = \frac{FAR + FRR}{2} \qquad (1)$$

The Root Mean Squared Error (RMSE) is a quadratic scoring rule which measures the average magnitude of the error. Expressing the formula in words, the difference between forecast and corresponding observed values are each squared and then averaged over the sample. Finally, the square root of the average is taken. The overall performance in this case was 91.31%.

The average recognition rate that has been established by using the same approach on **DB2** was of 92.75%. The classification rates for the different users of the DB2 database can be observed in figure 8.

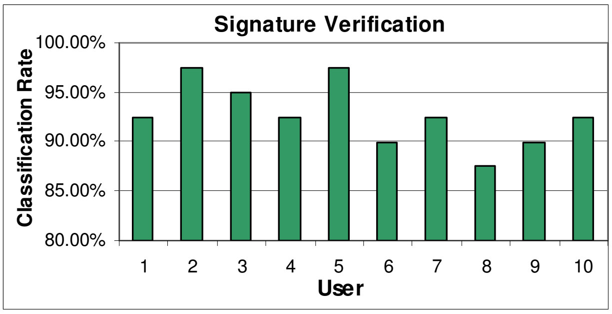

*Figure 8. Verification rates for the **DB2** enrolled users*

VI. CONCLUSIONS

Automatic signature recognition has been an intense research area because of the social and legal acceptance and widespread use of written signature as a personal authentication method. It is still is a challenging problem.

In this paper a new feature extraction scheme, based on TESPAR technique, has been suggested for online signature identification and verification. The TESPAR coding method allows generating from the time sequences that describe the local properties of the signature some global features stored in fixed dimensions matrices. A large number of candidate feature subsets were evaluated, such as: *Y, Vx, Vy, Pressure, Ax, Ay, number of Epochs* and that providing the best results were been selected. Our paper proposes and presents a new on-line signature verification and identification approach, based on the TESPAR coding method and SVM classifier. The results show that RBF Kernel outperforms Poly Kernel in both verification and identification tasks. WLSVM shows its remarkable advantages, in terms of memory usage and running performance.

The results obtained show that the proposed system is capable of identifying signatures from the **DB1** database with 84% accuracy and of verifying signatures from the **Task2-SVC2004** database with an average of 91.31% and with an average of 92.75% for **DB2**. These results were provided by the system when the fusion of TESPAR matrices generated from the selected features (*Y, Vx, Vy, Pressure, number of Epochs)* was employed. The fused vector dimension is made out of 112 coefficients.

We find the results to be promising taking into account the simplicity of the signature feature extraction and the fixed dimension of the matrices.

Due to the reduced number of epochs, many of TESPAR matrices coefficients are zero so they are not useful in the recognition task. For future developments, we intend to detect and remove all useless coefficients from the fused vector, so as to reduce its dimension. We will investigate new signature features as well, in order to improve the recognition results. The other facilities of WEKA and WLSVM applications will also be employed, in order to improve system performances.

ACKNOWLEDGEMENT

This work has been supported by the grant project CNCSIS No. 339/2007.

REFERENCES

[1] http://scgwww.epfl.ch/courses/
[2] http://ww.cse.msu.edu/~cse802/Papers/802_Signature_Verification.pdf
[3] A. K. Jain; P. Flynn, A. Ross, *Handbook of Biometrics*, Springer, 2008
[4] A.K. Jain, F.D. Griess, S.D. Connell. "On-line signature Verification" *Pattern Recognition*, 35(12), pp 2963–2972, Dec. 2002.
[5] .T. Ibrahim, Ling Guan **"**On-line signature verification by using most discriminating points", ICPR 2008. *19th International Conference on Pattern Recognition***,** pp 1-4,Tampa, 8-11 Dec.
[6] A.Khalmatov, B.Yanikoglu. "Identity authentication using improved online signature verification method"*Pattern Recognition letters* (2005), pp. 2400-2408
[7] R. A. King, T. C. Phipps. "Shannon, TESPAR and Approximation Strategies", *ICSPAT 98,* Vol. 2, pp. 1204-1212, Toronto, Canada, September 1998.
[8] E. Lupu**,** V.V.Moca, P.G. Pop "TESPAR coding study for speaker recognition*" The 30th session of scientific presentations "Modern technologies in the XXI Century"* pp. 214-221, *Bucharest 2003*
[9] http://www.wipo.int/pctdb/en/, U.S. Patent 6748354, 2004
[10] Dit-Yan Yeung, Hong Chang, et al., "SVC2004: First International Signature Verification Competition", *Proceedings of the International Conference on Biometric Authentication (ICBA)*, Hong Kong, 15-17 July 2004.
[11] Y. EL-Manzalawy, V. Honavar, WLSVM Integrating LibSVM into Weka Environment, 2005. Software available at http://www.cs.iastate.edu/~yasser/wlsvm
[12] Chih-Wei Hsu, Chih-Chung Chang and Chih-Jen Lin, "A Practical Guide to Support Vector Classification", *Department of Computer Science National Taiwan University, Taipei 106, Taiwan*
[13] http://www.dcs.shef.ac.uk/~fabien/publications/cardinaux_thesis.pdf