

An Off-Line Signature Verification System Using Hidden Markov

Model and Cross-Validation

EDSON J. R. JUSTINO¹
ABDENAÏM EL YACOUBI¹
FLÁVIO BORTOLOZZI¹
ROBERT SABOURIN^{1,2}

¹ PUCPR - Pontifícia Universidade Católica do Paraná, R. Imaculada Conceição, 1155 CEP:80215-901 - Curitiba - PR - Brasil

{justino,yacoubi,fborto}@ppgia.pucpr.br

² ETS - Ecole de Technologie Supérieure, 1100, rue Notre-Dame Ouest – Montréal (Québec) H3C 1K3 – Canada

sabourin@gpa.etsmtl.ca

Abstract. This work has as main objective to present an off-line signature verification system. It is basically divided into three parts. The first one demonstrates a pre-processing process, a segmentation process and a feature extraction process, in which the main aim is to obtain the maximum performance quality of the process of verification of random falsifications, in the false acceptance and false rejection concept. The second presents a learning process based on HMM, where the aim is obtaining the best model. That is, one that is capable of representing each writer's signature, absorbing yet at the same time discriminating, at most the intra-personal variation and the interpersonal variation. A third and last part, presents a signature verification process that uses the models generated by the learning process without using any prior knowledge of test data, in other words, using an automatic derivation process of the decision thresholds.

Introduction

In the last few decades, many approaches have been developed in the pattern recognition area, which approached the offline signature verification problem [2,3,4,5]. There are two main approaches for off-line signature verification: statistical approaches and pseudodynamic approaches [10]. The first one involves perceptive characteristics, therefore easy to imitate. The second involves imperceptive characteristics, therefore difficult to imitate [7].

As for the verification process, there are many approaches that are used nowadays, for example, Neural Networks [4,5], the Euclidean Distance Classifiers [3], Elastic Image Matching [12] and others. Hidden Markov Models have, in the last decades, attracted the attention of many researchers in the pattern recognition area, for example the recognition of handwritten text [6], the speech recognition [1] and recently the verification of on-line signatures [8]. These stochastic models have the capacity to absorb the variability between patterns and their similarities. Figure 1, which is a super-imposition of various specimens (previously skelitized) from the same author, shows some of these variability.

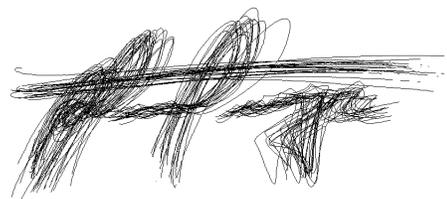


Figure 1 The super-imposed examples of the same writer's specimen skeletons.

The first objective of this work is to demonstrate the robustness of one simple static feature, or the density of pixels, when using it in an offline signature verification system complying to the false acceptance and false rejection concept, in the context of random signature falsifications. The second objective is to demonstrate the HMM's efficiency, using the cross-validation process to obtain the best representative signature model from each author. The third is to present a method for the definition of an automatic threshold in order to accept true and and reject false signatures. A set of 100 authors, making up a total of 4,000 signatures, are used in two distinct bases, one composed of 40 authors for the initial testing, and

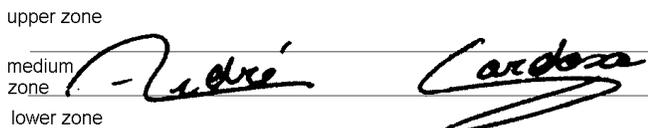
another consisting of 60 authors used to validate the generalization process.

2 The Feature extraction phase

The signatures were collected using either black or blue ink (no pen brands were taken into consideration), on a white A4 sheet of paper, with four signatures per page. A scanner subsequently digitized the four signatures, contained on each page, with a 300-dpi resolution in 256 grey levels. Afterwards the images were cut and pasted in a rectangular area of 3 x 10 cm or 400 x 1,000 pixels and were each saved separately in files. A group of 100 writers / authors were used to collect 40 specimens of each, making up a total of 4,000 signatures.

2.1 The horizontal segmentation strategy

The Graphometry divides the written area into 3, sections: upper zone, medium zone and lower zone [7,11]. The area of the medium zone is defined by the text main body part, the lower zone describes the descenders and the upper zone describes the ascenders. In signatures, this body of the text might not be as easily found since graphic signatures, see figure 2, may not contain the same characteristics of those in a common manuscript text. Therefore, defining the written zones in a signature may not be adequate when the purpose is identifying the ascendants, descendants and the medium zone, as it is in the common texts.



(a)



(b)

Figure 2 (a) Signature example using graphometric segmentation; (b) Rubric example.

For practical reasons, certain restrictions were chosen for the number of segmentation zones being used as well as for the number of pixels. The area destined to signatures was divided into four zones. All of which had the same sizes and didn't take into consideration the body of the signature as well as the upper and lower loops.

The use of multiple resolutions, has been adopted in the signature segmentation process [10]. This practice is guaranteed with the use of multiple classifiers in the verification process as well as the use of multiple features. One important aspect found in the use of this technique, is the ability of analyzing the signatures under several resolutions. One important factor that must be kept in mind is that for each feature used; the segmentation process has to be adjusted. For example, during the determination of the curvature of the segment, it is expected that this segmentation will be able to incorporate the angular aspects of the analyzed segment, which becomes very difficult to achieve when using a segmentation process that contains cells that are too small. In this case, the use of a segmentation process with larger cells is called for, that is, a more global evaluation. Starting from the four zones described earlier, the identification of the possible number of zones that could be used in the multiple resolution process was sought (in practice, all numbers that were whole dividers of 400). The group of possibilities can be seen in table 1.

Table 1 Horizontal Segmentation Table with Multiple Resolutions.

Cells Resolutions	Number of Vertical Cells	Pixel Number in Vertical Direction
Low	4	100
Low	5	80
Medium	8	50
Medium	10	40
Medium	16	25
Hight	20	20
Hight	25	16

The upper limit presented by table 1, 25 cells (16 pixels), was defined by using as reference twice the average width of the segments found in many writer's signatures from the database used (300 dpi and any kind of pen). This prevents a larger processing time cost, provoked by using excessively small cells in the segmentation. From the group of resolutions presented, 4

was chosen for being closer to the global analysis, 25 for high resolution analysis and 10 for representing an intermediate value between the two extremes.

2.2 The vertical segmentation strategy

The use of scales with square cells is common in signature verification [10][8]. This technique might not be the most adequate when using cells with an elevated size area (in this work this is called low resolution). In figure 3, it is possible to see the effect provoked by cells with a low resolution. In the specific HMM case, a very small number of observations do not adequately absorb all the details that would help to correctly describe a given signature model. Consequently, a 16-pixel segmentation was chosen, which represents twice the width of the traces of the signature and is the same used by the horizontal segmentation with 25 cells, see table 1. With this value, a minimum number of observation sequences would be guaranteed.

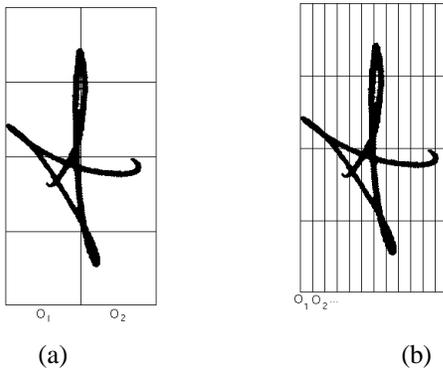


Figure 3 (a) Observation sequences, using square cells; (b) Observation sequences, using rectangular cells.

The difference in the horizontal direction resides in the variability of the number of cells, since it does depend on the length of the signature. These variations in the number of horizontal cells are also important when discriminating signatures from different writers. That is, there won't be great variations between specimens from the same writer, but there might be significantly large variations between different writers.

2.3 Graphometric features incorporated by the segmentation method

As seen in the prior discussed topic, the use of the scale makes it possible to incorporate each writer's personal characteristics, allowing the discrimination between two signatures that belong to different writers.

Besides the length of the signature, other characteristics, as the ones from graphometry can be implicitly observed with the use of the scale. This is possible due to the analysis of the occupied space destined to the signature. In the following, such characteristics are described.

2.3.1 Static features

Calibre: The calibre incorporates two important geometric characteristics in the global evaluation, which are the height and width of the signature. These characteristics are perfectly absorbed with the use of the scale. In figure 4, a difference in the cell occupation could be observed.

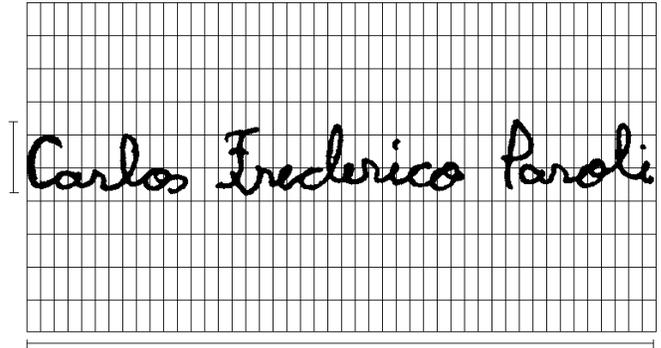


Figure 4 Examples of the Calibre incorporation graphometric features.

Proportion: The geometric regularity in the form of a signature is the main characteristic observed in the primitive proportion. With the scale it is possible to incorporate the regularity of the characteristics of writing, see figure 5. In this case, there is also a perfect relation with the occupation of the cells.

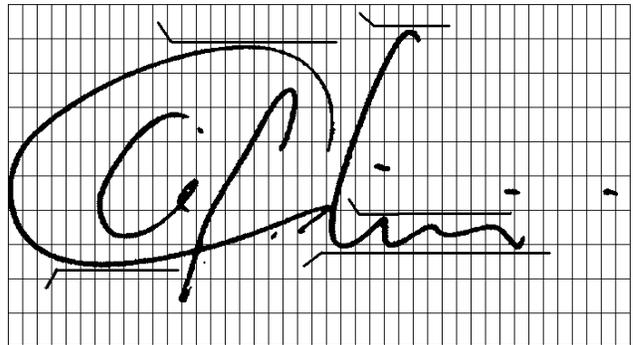


Figure 5 Example of the Proportion Incorporation.

Spacing: The spacing reveals the geometric behavior of the spaces that exist between the blocks of a signature, or

the simple lack of them, see figure 6. Figure 1 allows us to see the geometric stability of the spaces between blocks that exist in the same writer's different specimens of signatures. The cells are able to identify this characteristic;

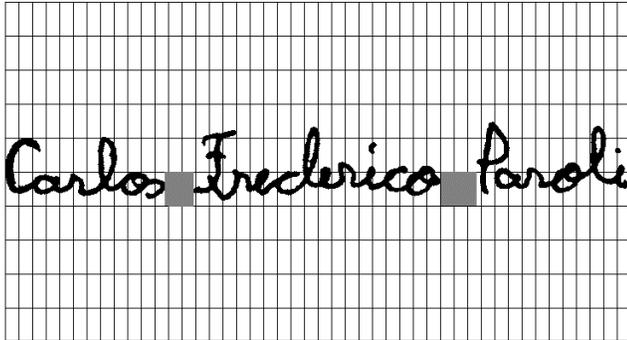


Figure 6 Example of the Spacing Incorporation graphometric feature.

Behavior guideline: it deals with the vertical movement of the signature according to the base line, it can't be analyzed without the presence or location of the base line. Since in the database used in this work, this information doesn't exist, this characteristic was deliberately omitted.

Base behavior: the base behavior describes the angle of inclination of the text during the writing elapse, according to an imaginary horizontal line. In figure 7, it is possible to see the process of the cell occupation due to this characteristic.

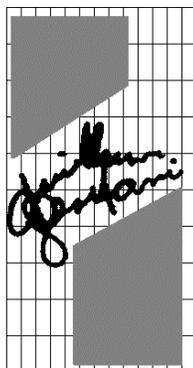


Figure 7 Examples of the base behavior variation.

2.4 The features

After the segmentation process, the feature extraction process starts. Each column of cells is converted into a characteristic vector, where each vector element has a representative numeric value. For this work only a static

feature was used, the density of pixels in each cell. The density of pixels has been frequently used by researchers and showed good results in the identification of the random falsifications, even though it doesn't have elements that incorporate the written dynamic aspects [8,12]. The good performance presented by this feature is easily justified because of the low intra-personal variability that exists between the various specimens belonging to the same writer, see figure 1.

To determine the density of pixels in each cell, the binary image of the signature over the scale was used and right afterwards the number of pixels was counted in each cell.

2.5 The symbol generation

The next phase, is the conversion of the vector group into a symbol sequence. To make this possible, the Vector Quantization – *VQ* process was used [8,9]. The first decision to be taken deals with the number of codebooks to be adopted. The first option is the use of one codebook for each writer.

As mentioned in [9], to obtain a good clustering it is desirable for each symbol or a codeword to be represented in the training set by a sufficient number of characteristic vectors, or more precisely at least two to five times the number of vector components used in clustering. If the number of vectors is too small, there is the risk of the non existence of a codebook that will be effectively representative. Therefore, a small number of training vectors requires a small codebook, what might not be sufficiently discriminative. Since the training database of the *VQ* in question works with a small number of specimens (30 specimens per writer, the learning and validation ones) we considered only one codebook for all the authors. For the *VQ*, the K-means algorithm was used.

3 Signature modeling using HMM

This section is divided into two parts. The first one, presents the cross-validation procedure used during the learning process. The second part presents the verification procedure.

3.1 The Learning Process

The main purpose of this phase is to generate an HMM $\lambda = \{A, B, \pi\}$ model that adequately characterizes each author signature model from the different writers. In this phase, the cross-validation procedure is used to optimally and dynamically define the optimal state number for each specific signature model (author model).

The correct choice of the model topology in HMM is fundamental to obtain a satisfactory result in the learning and verification phase. There are various topologies for the HMM models, each of which adapt to one particular characteristic. For the discrete models, two factors are predominant [1]. The first is the number of states to be used and the second is the number of transitions between these states. We chose a discrete left-to-right model, because it perfectly adapts to the dynamic characteristics of the Latin handwriting, in which the hand movements are always from left to right. Figure 8, shows the adopted topology for which no state skip is allowed. This topology is sufficient to model the variations of one writer's different specimens, as seen in figure 1.

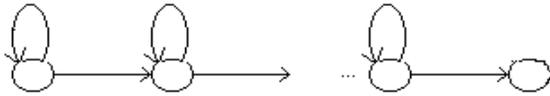


Figure 8 The Left-Right model used in the HMM Learning Process.

It is common practice to use a fixed number of states for the learning process. With signatures this isn't satisfactory, for the simple reason that each writer has a different signature size, that on one hand, requires a differentiated state level treatment for each writer, yet on the other hand, helps in the personalizing of the model [8].

Another determining factor for the good choice of a model is the use of cross-validation [6]. This procedure is part of the learning process, and has as main function, helping in the choosing of the model that has the best generalization capabilities over unseen data. The cross-validation permits choosing a model that is more adequate to each writer. Figure 9 shows the typical curves in a learning process using cross-validation. The logarithm's maximum point of validation probability $p_{vc}(O/\lambda)$, is used when choosing the most adequate model $p_t(O/\lambda)$ for one specific number of state. Hence, the choice of the best model was made basically due to the best results in terms of the probabilistic learning logarithm and also to the cross-validation, simultaneously.

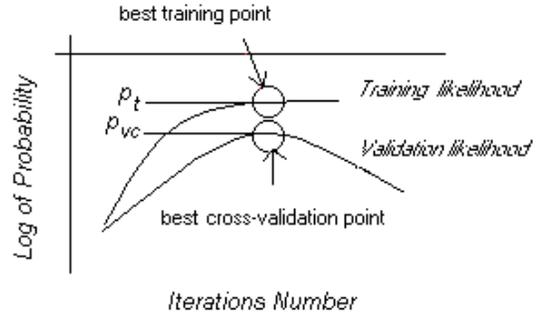


Figure 9 A typical learning and cross-validation curve for a set number of states.

In this phase too, we define the parameters that determine the acceptance and rejection borderlines, for a signature, α_1 and α_2 . The α_1 and α_2 values are defined after the termination of the model's $\lambda = \{A, B, \pi\}$ learning process. That is, when a given author's model has already been determined. The α_1 and α_2 that obtained the smallest average error rate on the validation, involving the type I error rates (false rejection) and type II (false acceptance), are chosen as additional parameters for the definition of the acceptance borderlines p_i and p_s , see figure 10.

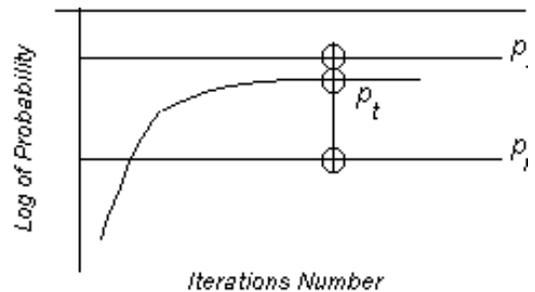


Figure 10 The borderlines used to delimit the area of acceptance and rejection in the validation process.

The medium threshold defined by the p_m , represent the learning probability logarithm, normalized by the observations number L , from the sequence in question. It is the reference for p_i and p_s parameters determination. The p_i normalization, generate a learning probability that has small variability between different samples probabilities.

$$p_m(O/\lambda) = \frac{\log p_t(O/\lambda)}{L}$$

$$p_i = p_m - (p_m \cdot \alpha_1)$$

$$p_s = p_m + (p_m \cdot \alpha_2)$$

3.2 The verification process

Before the verification of a signature, this one is transformed into a sequence of observations using our feature extraction scheme, described earlier. The verification process is basically made up of the Forward algorithm [1], used in this case to determine the logarithm of the probability of the observation sequence given the model of the claimed author. This log probability is then normalized by L .

$$p_{vn}(O/\lambda) = \frac{\log p_v(O/\lambda)}{L}$$

For the acceptance or rejection of a specimen the following criterion is used:

$$p_s \leq p_{vn} \leq p_i$$

4 The evaluation method

The database of 100 writers was divided into two parts, one containing 40 writers and the other containing the rest. The first database was used to carry out the first tests, to determine the VQ symbols, the α_1 and α_2 parameters and the best group of symbols and cells. The second was exclusively for the tests.

For the learning database 20 signatures were used after being chosen at random. Another 10 were used for the validation database and the last 10 were chosen for verification.

The performance of the system was evaluated using the average error rate, obtained by calculating the type I and type II average error rates, of all the writers that participated in the experiment. In other words, for each writer 10 real signatures were tested and the sum of the remaining participant writer's signatures was used as false.

In order to be able to evaluate the system behavior under different conditions, a group of scales with

different sized cells was used, the same occurred with the codebook. For the scale resolution, 4, 10 and 25 cells in the vertical direction were used. For the codebook, the values from 60 to 100 symbols with gaps of 10 were used.

5 The experiments

Two experiments were held. The first one used the first database (with 40 writers). The second used the remaining 60 writers. As mentioned before, the first database served for the definition of the parameters of the system.

In table 2, the experimental results obtained from the first database, earlier described, are presented. The results show the performance of the verification system, when speaking of the false rejection rate (error type I) and false acceptance rate (error type II). The column indicated as tests represents the group of signature types in testing. The first number that makes up the name of each group represents the number of vertical cells used. The second represents the size of the used codebook.

Table 2 Obtained results using the database containing 40 writers.

Tests	Error Type I (%)	Error Type II (%)	Mean Error (%)
A04_60	2.50	0.47	1.49
A04_70	3.25	0.37	1.81
A04_80	2.75	0.67	1.71
A04_90	3.50	0.29	1.89
A04_100	4.50	0.74	2.62
A10_60	1.50	0.36	0.93
A10_70	3.75	0.23	1.99
A10_80	1.75	0.24	0.99
A10_90	1.00	0.32	0.66
A10_100	1.25	0.29	0.77
A25_60	3.25	0.62	1.94
A25_70	2.50	0.67	1.58
A25_80	3.25	0.45	1.85
A25_90	2.00	0.42	1.71
A25_100	2.00	0.67	1.34

The three best results were used to determine the best number of cells (grid resolution) and the best symbol number. In so being, based on the results presented by the first database, the test was now held using the second database. It should be remembered that for the second test, the VQ and α_1 and α_2 base of determination was the same as the one in the first. This procedure is justifiable

because it permits a general use of its parameters. The results presented by the second base can be seen in table 3.

Table 3 Obtained results using the database containing 60 writers.

Tests	Error Type I (%)	Error Type II (%)	Mean Error (%)
A04_60	1.67	0.75	1.21
A10_90	2.17	1.23	1.70
A25_100	2.50	0.83	1.67

In a second experiment the obtained results by the three resolutions were combined in a classifier based on the Majority Vote Rule. In other words, each resolution scale provides a partial decision, the final decision being obtained by combining the partial decisions as shown in figure 11.

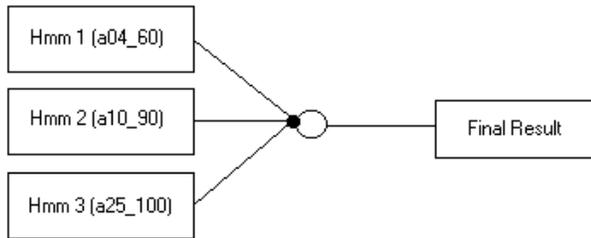


Figure 11 The combination of the three classifiers based on the HMM, using the Majority Vote technique.

In table 4, the combination results are presented, using the Majority Vote Rule, when applied to the best results obtained in the first and second bases.

Table 4 Combination results from the multiple resolutions, using two different databases.

Tests	Error Type I (%)	Error Type II (%)	Mean Error (%)
Database 1 (40 writers)	0,75	0,18	0,46
Database 2 (60 writers)	1,17	0,64	0,91

6 Conclusions

The main purpose of this work is to present a basic and robust system for the verification of static or offline signatures. For this, simple features were used so that the performance of the learning and verification system, using HMM, could be confirmed.

Another important factor lies in the low variability of the average error rate between the two bases used. This confirms the generalization capabilities of our approach when dealing with new authors. That is, the design of the VQ codebook as well as the derivation of the parameters α_1 and α_2 values, based on a limited group of writers causes only a 0,5% increase of the error rate on the second database for which no author contributed some signatures in the derivation of the parameters mentioned above.

In continuation to the present work, the next objectives are: To incorporate new features, other than the density of pixels so as the pseudo-dynamic characteristics described by graphometry can also be incorporated, like the curvature for the biggest stroke of each cell and the predominant stroke slant in each cell . Afterwards, we will combine these features in one unique HMM based classifier.

References

- [1] L. Rabiner, B. Juang, "Fundamentals of Speech Recognition", *Prentice Hall, New Jersey* (1993).
- [2] J. K. Guo, D. Doermann, A. Rosenfeld, "Local Correspondence for Detecting Random Forgeries" , *IEEE, 0-8186-7898-4* (1997).
- [3] Q. Yingyong, B. R. Hunt, "Signature Verification Using Global and Grid Features", *Pattern Recognition – vol 22, no. 12, Great Britain* (1994), 1621--1629.
- [4] K. Huang, H. Yan, "Off-line Signature Verification Based on Geometric Feature Extraction and Neural Network Classification", *Pattern Recognition – vol 30, no. 1, Great Britain*, (1997), 9--17.
- [5] R. Bajaj, S. Chaudhury, "Signature Verification Using Multiple Neural Classifiers", *Pattern Recognition – vol 30, no. 1, Great Britain* (1997), . 1--7.
- [6] A. El Yacoubi, M. Gilloux, R. Sabourin and C. Y. Suen, "Unconstrained Hand Written Word Recognition using Hidden Markov Models", *IEEE Transation on Pattern Analysis and Machine Intelligence, Vol. 2, no. 8, August* (1999), 752--760.
- [7] R. Sabourin, R. Plamondon, "Progress in the Field of Automatic Handwritten Signature Verification Systems Using Gray-level Images", *Inter. Workshop on Frontiers in Handwriting Recoginitin, Abril, Montreal* (1990).

[8] G. Rigoll, A. Kosmala, "A Systematic Comparison Between On-Line and Off-Line Methods for Signature Verification with Hidden Markov Models", *14th International Conference on Pattern Recognition – vol II, Australia* (1998), 1755--1757.

[9] A. J. Elms, "The Representation and Recognition of Text Using Hidden Markov Models", *Teses Doctor, University of Surrey, U.K.* (1996).

[10] R. Sabourin, G. Genest and F. Prêteux, "Off-line Signature Verification by Local Granulometric Size Distributions", *IEEE PAMI, VOL. 19, No. 9, september* (1997), 976--988.

[11] Justino, Edson J. R., F. Bortolozzi and R. Sabourin, "Os Sistemas de Reconhecimento de Assinaturas sob a Ótica da Grafologia", *ISDM'98, International Symposium of Document Manager – Curitiba - Brazil – ISSN – 0302 – 9743* (1998), 242--254.

[12] Bruyne, P. de and R. Forré, "Signature Verification with Elastic Image Matching", *International Carnahan Conference on Security Technology, Gothenburg, Sweden, August* (1986), 12--14.