OFF-LINE SIGNATURE VERIFICATION USING HMMS AND CROSS-VALIDATION

A. El-Yacoubi,[1] **E. J. R. Justino**[1], **R. Sabourin,**[2] and **F. Bortolozzi**[1]
[1] PPGIA, Pontificia Universidade Catolica do Parana (PUCPR)
Av. Imaculada Conceicao, 1155 - Prado Velho
80.215-901 Curitiba - PR - BRAZIL
e-mail: {yacoubi, justino, fborto}@ppgia.pucpr.br
[2] Ecole de Technologie Supérieure, Department of LIVIA
1100 Notre-Dame Ouest, Montréal, Canada H3C 1K3
e-mail: sabourin@gpa.etsmtl.ca

ABSTRACT

In this paper, we propose a new HMM-based approach for off-line signature verification. One of the novelty aspects of our method lies in the ability to dynamically and automatically derive the various author-dependent parameters, required to set an optimal decision rule for the verification process. In this context, the *cross-validation* principle is used to derive not only the best HMM models, but also an optimal acceptation / rejection decision threshold for each author. This leads to a high discrimination between actual authors and impostors in the context of random forgeries. To quantitatively evaluate the generalization capabilities of our approach, we considered two conceptually different experimental tests carried out on two sets of 40 and 60 authors respectively, each author providing 40 signatures. The results obtained on these two sets show the robustness of our approach.

1. INTRODUCTION

Signature verification is one of the most important research areas in the field of person authentication by biometric techniques. Its potential applications are numerous and include, personal verification for access control, banking applications, electronic commerce, etc. Two categories of verification systems are usually distinguished: on-line systems [9] for which the signature signal is captured during the writing process, thus making the dynamic information available, and off-line systems [11] for which the signature is captured once the writing process is over, and thus only a static image is available. As a matter of fact, off-line signature verification is a much more difficult problem, since many desirable author-sensitive characteristics such as the order of strokes, the velocity and other dynamic information are not available in the off-line case. There are two types of forgeries involving the signature verification task: random forgeries, i.e. the impostor has no knowledge of

the signature of the actual author, and skilled forgeries, i.e. the impostor has this knowledge, and thus he is able to more or less imitate the signature. Although, random forgeries are less difficult to reject than skilled forgeries, their consideration is of equal importance since it is believed that more than 90% of the forgeries observed in real life are random.

In this paper, we propose a hidden Markov model (HMM) based approach for off-line signature verification in the context of random forgeries. Even though HMMs have been extensively used in handwriting recognition in the last decade [2] [3] [4] [6], their use in signature verification tasks is still limited, whether in the on-line case [7] [8] [12], or in the off-line case [10]. Our approach is similar to those mentioned above in the fact that the set of signatures of each author is assumed to be produced by an ad hoc HMM. In the verification process, a given signature is accepted as belonging to the claimed author or rejected (belonging to an impostor) depending on whether the matching score or the likelihood of the signature generated by the associated HMM is above a preset threshold. The novelty of our approach lies in the setting of such a threshold, which is, in our case, obtained dynamically for each author by minimizing the associated error rate on a subset of the training set, called the *validation* set, and used to simulate the actual verification process. This is known as the *cross-validation* principle. Moreover, the threshold for a given author does not depend only on the distribution of the likelihoods of the signatures pertaining to this author, but also on the "distance" between the distribution of these likelihoods and the distribution of the likelihoods of the signatures of the impostors. The latters are considered as the signatures of a subset of the other authors existing in the training phase.

To evaluate the generalization capabilities of our approach, we consider two conceptually different experimental tests carried on two sets of 40 and 60 authors respectively, each author providing 40 signatures. The difference between these two sets lies in the fact that a subset of the first one is used the represent the *impostor* space for any tested author. The second test does not provide to the verification system a similar subset and hence the associated error rate is expected to be higher than the “optimistic” error rate associated with the first set. Therefore, the difference between the two error rates can give us a good estimation of the robustness and generalization capabilities of the proposed system when tested on new authors.

To characterize the signature images, we currently consider simple features consisting of vectors, the components of which are the pixel densities computed on local square cells. These vectors are then converted into discrete symbols using vector quantization. As will be shown in this paper, these features, as used in our HMM framework, are robust in detecting random forgeries.

The organization of this paper is as follow. Section 2 describes the databases used in our experiments. Section 3 deals with signature image characterization. Sections 4 and 5 detail the HMM modeling of signatures and the design of the verification algorithm respectively. Section 6 deals with the experiments carried out to validate the proposed approach. Finally, Section 7 gives a conclusion and discusses some perspectives of the proposed approach.

2. DATABASES

As pointed out earlier, we use two databases for our experiments. The first one, called DB I, consists of 40 authors, while the second one, called DB II, consists of 60 authors. Each author provided 40 signatures, among which 30 are used for learning purposes and 10 for *verification*. The first set of 30 signatures was further divided in two sets: the actual *training* set, and a *validation* set used to simulate the verification process. The difference between DB I and DB II is that the training and validation sets of DB I are used to simulate (or represent) the entire *impostor* space in the training phase. DB I also serves for constructing the VQ codebook as shown in the next section.

The signatures were written on white sheet of paper, using any type of blue or black pen, and were scanned at a resolution of 300 *dpi* and binarized using the algorithm described in [1]. Table 1 recalls the size of the training, validation and test sets for DB I and DB II.

Table 1. The Databases used in the experimental tests

Set	Number of authors	Training set per author	Validation set per author	Test set per author	Total
DB I	40	20	10	10	1600
DB II	60	20	10	10	2400

3. REPRESENTATION OF SIGNATURE IMAGES

Our verification system uses simple features to capture the individual characteristics of signatures. First, a grid is superimposed on the signature image, segmenting it into local square cells (Figure 1). From each cell, the *pixel density* is computed and considered as the associated local feature. Thus, the signature image is represented by a sequence of cell columns. Associated to each column is a real vector, the components of which are the cell pixel densities. Various sizes of square cells are experimented: 100x100, 40x40 and 16x16 pixels. Given that the allowed height of a signature image is 400 pixels (typical height of a Brazilian check), this leads to consider vectors of dimension 4, 10 and 25 respectively. The aim of using various cell sizes is to be able to analyse the signatures under several scales or resolutions: 100x100 and 16x16 allow us to analyse the signature signal at low resolution (global information) and high resolution (small details) respectively, and 40x40 pixels correspond to an intermediate resolution allowing us to make a compromise between these two extremes.

Note that, here, the whole vertical area allowed for signatures is used instead of the sole signature signal. This has the advantage to take the vertical location of the signature into account, thus leading to a better characterization of author inter-variations (variations between signatures of different authors).

Figure 1. Segmentation of a signature into local square cells

For each cell size, the signature image is transformed into a sequence of real vectors, which are further transformed into a sequence of discrete symbols using the *k-means* vector quantization (VQ) algorithm. Ideally, it is desirable to derive one codebook for each author in order to add a more author-sensitive characteristic in the verification process [10]. However, given the small amount of training signatures available for each author, we observed that this strategy leads to poor performance since a large mismatch is observed between the vector-quantized data of the training and test sets. To overcome this problem, we derive one codebook for all the possible authors using all the training signatures of the DB I set (40 x 20 = 800 signature images). The size of training vectors becomes, in this way, much larger, thus allowing a more reliable estimation of the codebook. By doing this, we hope that the vectors extracted from the whole DB I training set (considering the 40 authors) will have a large coverage of the entire vector space, including the vectors belonging to signatures of authors possibly not considered in training the codebook. This hypothesis is reasonable, since the training vectors independently (i.e. without taking into account the signatures they are extracted from) participate to the estimation of the codebook (memoryless VQ). In section 6, we will show the validity of this approach by analysing the results obtained on the DB II set, the signatures (authors) of which were not used in training the codebook.

4. HMM MODELING OF SIGNATURES

The advantage of modeling signatures by HMMs lies in their ability to capture the local characteristics in the square cells and in their flexibility to model the variability of signature shapes and lengths. Also, HMM outputs are probabilities and hence they are particularly suitable for implementing the acceptation / rejection mechanism required in the verification process. The model we consider for each author has a simple left-to-right structure with no state skip allowed (Figure 2).

This structure is convenient to build author-dependent models for which temporal distortions are not as high as in an omni-scriptor handwriting recognition task for instance. Moreover, since no state skip is allowed, the number of HMM parameters is reduced which is a very desirable property given the small amount of available training data for each author.

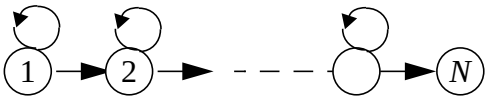

Figure 2. The structure of the HMM model considered for each author

To obtain the optimal model for each author, we make use of the associated training and validation sets. The choice of the number of states and the training of each model is performed using the *Baum-Welch* algorithm and the *cross validation* principle: the optimal number of states and the associated optimal HMMs parameters are those yielding the highest likelihood on the validation set. This allows an appropriate and optimal choice of the number of states for each author while preventing an overlearning of the training set. In this guaranteed, a good generalization of the trained model over unseen signature samples is obtained. Typically, the number of states for each author depends on the average length of its training observation sequences and on the range of author intra-variations.

5. THE DESIGN OF THE VERIFICATION PROCESS

The goal of signature verification is to automatically accept or reject the identity claimed by the author. To evaluate a signature verification system, two types of error rates are usually defined for each author: *False Rejection Rate* (*FRR*) or *Error Type I*, i.e. the percentage of rejection of genuine signatures and *False Acceptance Rate* (*FAR*) or *Error Type II*, i.e. the percentage of acceptance of forgeries. In our system, we consider random forgeries consisting of the test signatures belonging to the remaining existing authors. The *author error rate* considered in our experiments is the average of FRR and FAR, and the *mean error rate* is the average of the error rates associated with existing authors.

It is worth to note that unlike in pattern recognition tasks, the goal in signature verification is not to seek the model with the highest matching score; but rather to check whether the matching score, associated with the signature claimed by an author, is *close* to the matching scores associated with the training signatures associated with the same author. Therefore, it is more reliable to use the same procedure in training as in verification to compute the matching scores. Hence, we compute the likelihood of a signature observation sequence in the verification (or validation) phase by the *Forward* algorithm rather than by the *Viterbi* algorithm, which is usually used in recognition tasks.

Another issue in using HMMs for signature verification is the *normalization* of the HMM outputs before the application of an acceptance / rejection decision rule.

This is required, because observation sequences corresponding to signatures of the same author typically have variable lengths. Since HMM outputs decrease exponentially when sequence lengths increase, it is not reasonable to directly use HMM outputs for verification purposes. In our system, the normalization is carried out by dividing the log probability of an observation sequence log$P$ by its length T:

$$P_n = (\log P) \ / \ T \qquad (1)$$

which is equivalent to considering the matching score of an observation sequence as the geometric average of the probability of one observation. This is can be viewed as an implicit normalization of the signature width. Once the normalization is done, a signature is accepted if the following decision rule is satisfied:

$$P_n > P_m^t - (w \cdot \left| P_m^t - P_m^i \right|) \qquad (2)$$

where $P_m{}^t$ is the expected probability of P_n, estimated as the mean of the probabilities (in the log domain and after normalization) of the training signatures of the author under verification, and $P_m{}^i$ is the mean of the probabilities of the training signatures of the other authors (the training impostor set). The goal of incorporating $P_m{}^i$ is to characterize the space of potential impostors for each author by exploiting the signatures of the other authors considered in training. This gives us the flexibility of appropriately choosing a decision threshold, depending not only on the expected probability $P_m{}^t$, but also on the distance between $P_m{}^t$ and $P_m{}^i$. This distance characterizes the author intra-variations as well as the author inter-variations. Indeed, small author intra-variations mean that the signatures are subject to less variation, making their training easier and their likelihood higher accordingly. On the other side, large author inter-variations mean that the signatures of the other authors have a quite different shape from those of the considered author. Hence their likelihood, given the model of the latter are likely to be low.

The *weight* parameter w (taking values in the set {0.0, 0.1, 0.2, ..., 1.0}) is also specific for each author and is used to minimize the associated mean error rate. It is also estimated using the cross validation concept, and corresponds to the value that minimizes the error rate on the validation set (rather than on the training set). Again, this prevents the derived w parameters, and consequently the author-dependent decision thresholds, from being highly correlated to the likelihoods of the training signatures. Note that, here, the 10 signatures corresponding to the author validation set are used to evaluate the *FRR* and the union of the validation sets of the other authors (the validation impostor set) is used to evaluate the *FAR*. It is important to note that the training and validation impostor sets, described above and used to estimate the author-dependent parameters $P_m{}^i$ and w, are constructed by the signatures of the DB I set only (see Section 2). This makes it possible to carry out two conceptually different experimental tests as will be explained in the next section.

6. EXPERIMENTS AND RESULTS

Our verification system was tested on the DB I (40 authors) and DB II (60 authors) test sets. Since 10 test signatures are considered for each author, the size of the DB I and DB II tests are 400 and 600 respectively. The experiments carried out on these two tests are conceptually different if we bear in mind the type of random forgeries the system has to deal with in each case. For both cases, the random forgeries for a given author are considered as the test signatures of the other authors (39 and 49 authors remaining in DB I and DB II respectively). However, in the DB I test, the system has an a priori knowledge of the random forgeries since the parameters P_m^i and w are optimized for each author using the impostor validation set of DB I. In the DB II test, the system does not have such a priori knowledge since the parameters P_m^i and w are estimated for each author in DB II using the same impostor validation set of DB I and not of DB II.

As discussed before, to be able to analyse the signature signal under several resolutions, we experimented several sizes of square cells: 100x100 pixels (low resolution), 40x40 pixels (intermediate resolution) and 16x16 resolutions (high resolution). For each resolution, we considered various VQ codebook sizes. Then, for each cell size / codebook size pair, we ran our verification system to evaluate the error rates. Table 2 shows these error rates computed on the DB I test set.

Table 2. GLOBAL ERROR RATE FOR VARIOUS CELL / CODEBOOK COMBINATIONS FOR THE DB I TEST

Cell Size	Codebook Size	Error Type I (%)	Error Type II (%)	Mean Error Rate (%)
100 x 100	**60**	**2.50**	**0.47**	**1.49**
100 x 100	70	3.25	0.37	1.81
100 x 100	80	2.75	0.67	1.71
100 x 100	90	3.50	0.29	1.89
100 x 100	100	4.50	0.74	2.62
40 x 40	60	1.50	0.36	0.93
40 x 40	70	3.75	0.23	1.99
40 x 40	80	1.75	0.24	0.99
40 x 40	**90**	**1.00**	**0.32**	**0.66**
40 x 40	100	1.25	0.29	0.77
16 x 16	60	3.25	0.62	1.94
16 x 16	70	2.50	0.67	1.58
16 x 16	80	3.25	0.45	1.85
16 x 16	90	3.00	0.42	1.71
16 x 16	**100**	**2.00**	**0.67**	**1.34**

As shown in Table 1, for every cell size, there exits an associated codebook size for which the global error rate is minimized. Since we are analysing the signature signal with low, intermediate and high resolution, it can be quite interesting to combine the results obtained with these three resolutions. Our method to perform this combination is to adopt a *majority vote rule* by considering for each resolution, the codebook size yielding the smallest mean error rate. This means to select the (100x100 / 60), (40x40 / 90) and (16x16 / 100) cell size / codebook size pairs as shown in bold style in Table 2. The result of this combination is shown in Table 3.

Table 3. MAJORITY VOTE RULE RESULT FOR THE DB I SET

Test Set	Error Type I (%)	Error Type II (%)	Mean Error Rate (%)
DB I	0.75	0.18	0.46

From Table 3, it is clear that the majority vote rule allows us to significantly reduce the mean error rate. This can be explained by the fact that this combination allows the verification system to analyse the signature signal under several resolutions before making a decision.

In the second phase of our experiments, we ran our verification system, in the same way, on the DB II set. However as DB II is used to simulate the integration of new authors in the verification system, we do not consider all the cell / codebook pairs as before and then select the optimal ones for combination. Rather, we ran the system with only the optimal ones selected a posteriori in the DB I test. This also will permit to evaluate the generalization capabilities of our methodology. Table 4 gives the results obtained with the 3 selected cell / codebook pairs on the DB II set and Table 5 shows the result of their combination by the majority vote rule.

Table 4. GLOBAL ERROR RATE FOR THE SELECTED CELL / CODEBOOK COMBINATIONS FOR THE DB II TEST

Cell Size	Codebook Size	Error Type I (%)	Error Type II (%)	Mean Error Rate (%)
100 x 100	60	1.67	1.14	1.40
40 x 40	90	2.17	1.56	1.86
16 x 16	100	2.50	1.08	1.79

Table 5. MAJORITY VOTE RULE RESULT FOR THE DB II SET

Test Set	Error Type I (%)	Error Type II (%)	Mean Error Rate (%)
DB II	1.17	0.64	0.91

Table 4 and Table 5 show that overall, the error rates obtained on the DB II set remain low even if they are slightly higher than those obtained on the DB I set. This was expected since 1) the VQ codebook is constructed from the DB I data only; 2) the parameters related to the decision threshold for each author (whether in DB I or in DB II) are optimized on a subset of DB I only and 3) the selection of the optimal codebook size for each resolution is carried out a posteriori on the DB I test only (the selection of these codebooks on the DB II would likely have yielded better results than those reported in Table 5). The fact that there is only such a slight increase in the errors rates for DB II demonstrates the robustness of our verification system when dealing with new authors.

It is hard to compare the results of our system with those reported by other approaches, due to the difference in the sizes and types of the databases used. For instance, the error rates on DB I and DB II compare favourably with the 1.9% error rate obtained with the HMM based approach reported in [10], and tested on a database of 14 authors, for which smaller training data and different forgeries are used but in the context of much more clean signatures since these signatures were actually acquired on-line and then transformed off-line. Hence, there was no binarization or noise problem as this often occurred in our case. Our results also compare favourably with the 1.1% error rate obtained by the local correspondence approach reported in [5], and tested on a database of 20 authors, the signatures of which are of text type only.

7. CONCLUSION

We described in this paper a robust HMM-based approach for off-line signature verification. By simulating, in the training phase, the actual verification process, the author-dependent thresholds needed to set the decision rule, were dynamically and automatically derived for each author considered in the enrolment phase. The results achieved are very promising, especially if we bear in mind the simple nature of the features used, and the fact that our system deals with text type signature as well as graphic text signatures (which are much more subject to distortions).

A straightforward improvement of our system, therefore, is the extraction of more discriminative features from signature images. We have already implemented the extraction of various types of geometric and structural features. The evaluation of their discriminative power is currently under investigation. Our future work will also be devoted to the automatic derivation of the optimal scale (cell size) for each author prior to verification, and to the investigation of other decision rules and more suitable decision thresholds to optimize the verification task. Finally, we will test our verification system in the context of skilled forgeries. In this case, the main adaptation of the current system is related to the feature extraction phase, by looking for pseudo-dynamic characteristics.

REFERENCES

[1] A. S. Abutaleb, "Automatic Thresholding of Gray-Level Pictures Using Two Dimensional Entropy", **Computer Graphics & Image Processing**, VOL. 47, NO. 1, pp. 22-32, 1989.

[2] M.Y. Chen, A. Kundu, and S.N. Srihari, "Variable Duration Hidden Markov Model and Morphological Segmentation for Handwritten Word Recognition," **IEEE Transactions on Image Processing**, Vol. 4, No. 12, pp. 1675-1687, Dec. 1995.

[3] A. El-Yacoubi, M. Gilloux, R. Sabourin, and C. Y. Suen, "Unconstrained Handwritten Word Recognition using Hidden Markov Models," **IEEE Transactions on Pattern Analysis and Machine Intelligence**, VOL. 21, NO. 8, pp. 752-760, August 1999.

[4] A. El-Yacoubi, R. Sabourin, M. Gilloux and C.Y. Suen, "Improved Model Architecture and Training Phase in an Off-line HMM-based Word Recognition System," presented at the 14th International Conference on Pattern Recognition, pp.1521-1525, Brisbane, Australia, August 16-20, 1998.

[5] J. K. Guo, D. Doermann and A. Rosenfeld, "Local Correspondence for Detecting Random Forgeries," Presented at the International Conference on Document Analysis and Recognition, pp. 319-323, Ulm, Germany, August 18-20, 1997.

[6] Ha, J.Y., Oh, S.C., and Kim, J.H., "Recognition of Unconstrained Handwritten English Words With Character and Ligature Modeling," **International Journal on Pattern Recognition and Artificial Intelligence**, Vol. 9, No. 3, pp. 535-556, 1995.

[7] R. Kashi, J. Hu, W.L. Nelson, and W. Turin, "A Hidden Markov Model Approach to Online Handwritten Signature Verification", **International Journal on Document Analysis and Recognition**, No.1, pp. 102-109, 1998.

[8] R. Martens, L. Claesen, "Incorporating Local Consistency Information into the On-line Signature Verification Process", **International Journal on Document Analysis and Recognition**, No.1, pp. 110-115, 1998.

[9] R. Plamondon, "The Design of an On-Line Signature Verification System: From Theory to Practice," **International Journal on Pattern Recognition and Artificial Intelligence**, Vol. 8, pp. 795-811, 1994.

[10] G. Rigoll, and A. Kosmala, "A Systematic Comparison Between On-Line and Off-Line Methods for Signature Verification with Hidden Markov Models", presented at the 14th International Conference on Pattern Recognition, pp. 1755-1757, Brisbane, Australia, August 16-20, 1998.

[11] R. Sabourin, G. Genest, and F. J. Preteux, "Off-Line Signature Verification by Local Granulometric Size Distributions,' **IEEE Transactions on Pattern Analysis and Machine Intelligence**, VOL. 19, NO. 9, pp. 976-988, September 1997.

[12] L. Yang, B. K. Widjaja, and R. Prasad, "Application of Hidden Markov Models for Signature Verification", **Pattern Recognition**, Vol. 28, No. 2, 1995, pp. 161-170.