Applying Dissimilarity Representation to Off-line Signature Verification

Luana Batista, Eric Granger and Robert Sabourin
Laboratoire d'imagerie, de vision et d'intelligence artificielle
École de technologie supérieure, Montreál, Canada
lbatista@livia.etsmtl.ca, {eric.granger, robert.sabourin}@etsmtl.ca

***Abstract*—In this paper, a two-stage off-line signature verification system based on dissimilarity representation is proposed. In the first stage, a set of discrete *left-to-right* HMMs trained with different number of states and codebook sizes is used to measure similarity values that populate new feature vectors. Then, these vectors are input to the second stage, which provides the final classification. Experiments were performed using two different classification techniques – AdaBoost, and Random Subspaces with SVMs – and a real-world signature verification database. Results indicate that the performance is significantly better with the proposed system over other reference signature verification systems from literature.**

***Keywords*-Off-Line Signature Verification; Dissimilarity Representation; Hidden Markov Models; AdaBoost; Support Vector Machines; Random Subspaces**

I. Introduction

Signature Verification (SV) systems are relevant in many real-world applications, such as check cashing, credit card transactions and document authentication. In off-line SV, handwritten signatures are available on sheets of paper, which are later scanned in order to obtain a digital representation. Given a digitized signature, an off-line SV system will perform preprocessing, feature extraction and classification (also called verification) [1].

Among several classification methods used in off-line signature verification (SV), the *left-to-right* topology of discrete Hidden Markov Model (HMM) [2] is known to adapt perfectly to the dynamic characteristics of the western handwriting [3], in which the hand movements are always from left to right. The traditional SV approach consists of training a HMM with only genuine signatures. Then, the decision threshold between the genuine and impostor's classes is defined by using a validation set that contains samples from both classes. Input patterns are assigned to the genuine class if its likelihood is greater than the decision threshold.

In contrast to this traditional approach, Bicego et al. [4] proposed a system for 2D-shape/face recognition in which both the genuine subspace, w_1, and the impostor's subspace, w_2, are modeled. Based on dissimilarity representation (DR)[1], their approach consists of using a set of continuous HMMs, trained with a fixed number of states, to extract similarity measures that define a new input feature space. The fact that two sequences O_i and O_j present similar degrees of similarity with respect to several HMMs enforces the hypothesis that O_i and O_j belong to the same class [4].

In this paper, a two-stage off-line SV system inspired by Bicego's DR concept is proposed. In the first stage, a set of discrete *left-to-right* HMMs, trained with different number of states and different codebook sizes, is used to measure similarity values that form new feature vectors. Then, these vectors are input to the classification stage, which provides the final decision. Since a very large number of HMMs are produced, an important aspect of this work is the selection of a subset of representative features in order to improve performance. Given its ability to perform feature selection while classifying, Adaboost [6] is employed in the classification stage. This is compared to a SV system that uses Random Subspaces [7] and Support Vector Machines (SVMs) for classification.

Experiments are performed with the Brazilian SV database [8] with random, simple and skilled forgeries. To analyze system performance under different operating points (i.e., thresholds), an overall ROC curve is generated. This curve also allows the system to dynamically select the most suitable solution for a given input pattern. This property is useful in banking applications, for example, where the decision to use a specific operating point may be associated with the amount on checks. The paper is organized as follows. The next section presents the proposed approach. Then, Section III describes the experimental methodology and Section IV presents and discusses the experiments.

II. A SV system based on DR

In this section, a two-stage off-line SV system inspired by Bicego's DR concept [4] is proposed. In the first stage, a set of HMMs, representing both genuine and impostor's classes, is used as a feature extractor to obtain similarity measures (likelihoods) that populate new feature vectors. These vectors are then used to train a two-class classifier, in the second stage, in order to find the decision boundary between genuine and impostor's classes.

The first stage is formally defined as follows. Let $w_1 = \{\lambda_1^{(C_1)}, ..., \lambda_R^{(C_1)}\}$ be the set of R representative models of the genuine class C_1; $w_2 = \{\lambda_1^{(C_2)}, ..., \lambda_S^{(C_2)}\}$ be the set of S representative models of the impostor's class C_2;

[1]In dissimilarity representation, an input pattern is described by its distances with respect to a predetermined set of prototypes [5]

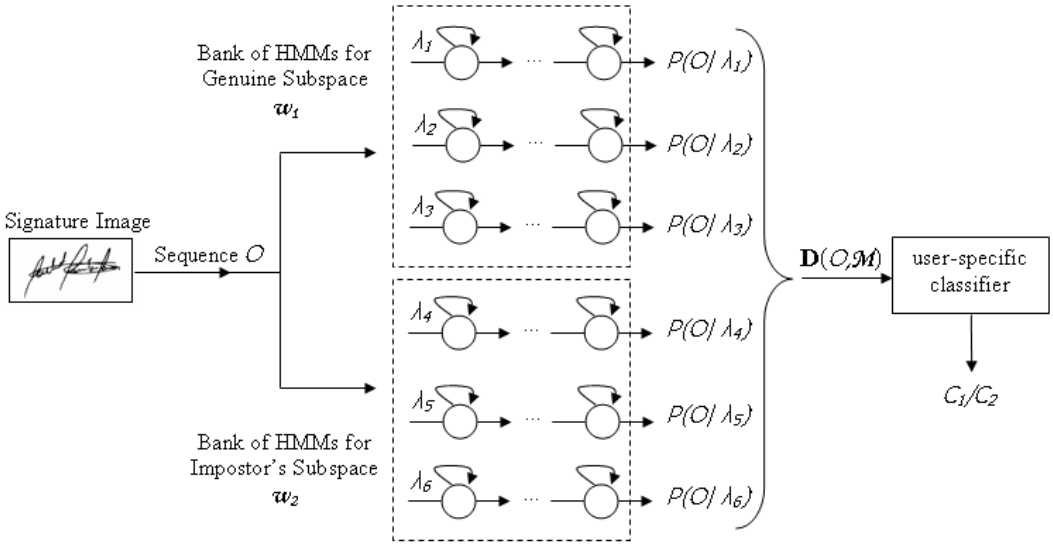


Figure 1. Block diagram of the proposed system using three HMMs per subspace ($S = R = 3$).

and $\mathcal{M}$ be the vector containing the representative models of both classes, that is, $\mathcal{M} = w_1 \cup w_2$. Given a training sequence $O_{trn} \in \{C_1|C_2\}$, its feature vector $\mathbf{D}\,(O_{trn}, \mathcal{M})$ is composed of the likelihoods computed between O_{trn} and every model in $\mathcal{M}$, that is,

$$\mathbf{D}\,(O_{trn}, \mathcal{M}) = \begin{bmatrix} P(O_{trn}/\lambda_1^{(C_1)}) \\ \cdots \\ P(O_{trn}/\lambda_R^{(C_1)}) \\ \\ P(O_{trn}/\lambda_1^{(C_2)}) \\ \cdots \\ P(O_{trn}/\lambda_S^{(C_2)}) \end{bmatrix}$$

In contrast to the Bicego's system, the HMMs in $\mathcal{M}$ are trained by using different number of states and different codebook sizes, as described in next section.

After applying the same process to all training signatures from C_1 and C_2, each feature vector $\mathbf{D}\,(O_{trn}, \mathcal{M})$ is labeled according to the class of O_{trn} and used to train a user-specific classifier[2] (or an ensemble of user-specific classifiers) in the second stage. During the test phase, the feature vector $\mathbf{D}\,(O_{tst}, \mathcal{M})$ is calculated for a given input sequence O_{tst}, and then sent to the classification stage, which takes the final decision.

Fig. 1 illustrates an exemple of the proposed system, where three HMMs per subspace are used. Observe that, if O_{tst} belongs to class C_1, the feature vector $\mathbf{D}\,(O_{tst}, \mathcal{M})$ should contain higher values in the first R positions and smaller values in the remaining S positions, which allows to discriminate between the classes C_1 and C_2. In a feature-based approach, O_{tst} would be assigned to the class of the most similar model. However, this approach does not use all the information contained in a dissimilarity space [4].

[2]The term *user-specific classifier* is used to differentiate from systems where a same *global classifier* is shared by all users.

III. Experimental Methodology

The Brazilian SV database [8] is used for proof-of-concept computer simulations. It contains 7920 samples of signatures that were digitized as 8-bit greyscale images over 400X1000 pixels, at resolution of 300 dpi. Three types of forgeries are considered in this database: random, simple and skilled. The random forgery is a genuine signature sample belonging to a different writer. It is produced when the forger has no access to the genuine samples, not even the writer's name. In the case of simple forgeries, only the writer's name is known. Thus, the forger reproduces the signature in his/her own style. A simulated forgery represents a reasonable imitation of a genuine signature.

The signatures were provided by 168 writers and are organized in two sets: the development database (DB_{dev}) and the exploitation database (DB_{exp}). DB_{dev} is composed of 4320 genuine samples supplied by 108 individuals, and it is used for designing codebooks and to train the HMMs that will compose the impostor's subspace, w_2. DB_{exp} contains 60 writers, each one with 40 samples of genuine signatures, 10 samples of simple forgery and 10 samples of skilled forgery. 20 genuine samples are used for training, 10 genuine samples for validation, and 30 samples for test (10 genuine samples, 10 simple forgeries and 10 skilled forgeries). Moreover, 10 genuine samples are randomly selected from the other 59 writers and used as random forgeries to test the current user-specific classifier. Each writer in DB_{exp} will, therefore, be associated to a genuine subspace, w_1.

The signature images are represented by means of density of pixels, extracted through a grid composed of cells of 40x16 pixels [9], as shows Fig. 2. For each writer from both DB_{dev} and DB_{exp}, a set of discrete HMMs is trained by using the *left-to-right* topology [2] with 20 genuine samples. 29 different codebook sizes (obtained by varying the number of clusters from 10 to 150, in steps of 5) and different number of states are used to generate the set of candidate HMMs. The maximum number of states is given by the smallest sequence of observations used for training. The genuine subspace, w_1, is therefore composed of a variable number of available HMMs, depending on the writer's signature size. On the other hand, to compose w_2, there are always 24505 available models taken from the 108 writers in DB_{dev}.

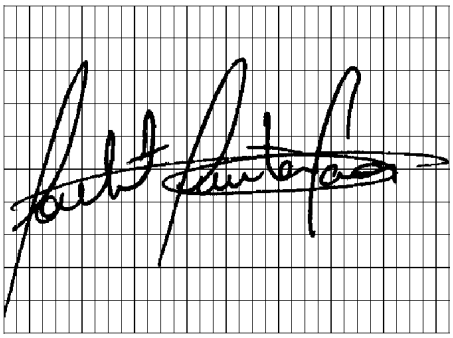


Figure 2. Example of grid segmentation scheme.

A common way to report the SV system performance is in terms of error rates. The false negative rate (FNR) is related to genuine signatures which were misclassified as forgeries. Whereas the false positive rate (FPR) is related to forgeries which were misclassified as genuine signatures. FNR and FPR, also known as type 1 and type 2 errors, respectively, can be used to generate a pair $\{TPR, FPR\}$ in the ROC space, since TPR = 1 - FNR. In this paper, it is assumed that the overall system's performance is measured by an averaged ROC curve obtained from a set of user-specific ROC curves using a validation dataset. The averaging method [10] generates an overall ROC curve taking into account user-specific thresholds. At first, the cumulative histogram of random forgery scores of each user i is computed. Then, the similarity scores (thresholds) providing a same value of cumulative frequency, γ, are used to compute the operating points $\{TPR_i(\gamma), FPR_i(\gamma)\}$. Finally, the operating points associated with a same γ are averaged. Note that γ can be viewed as the true negative rate (TNR = ratio of negatives (forgeries) correctly classified to the total of negatives) and that it may be associated with different thresholds. Fig. 3 shows an example where the thresholds associated with $\gamma = 0.3$ are different for users 1 and 2, that is $t_{user1}(0.3) \cong -5.6$ and $t_{user2}(0.3) \cong -6.4$.

To measure the system's performance during test, false negative rates and false positive rates are calculated by using the user-specific thresholds associated to different γ values of the averaged ROC curve. The average error rate (AER), also computed for different γ, indicates the total error of the system, where FNR and FPR are averaged taking into account the *a priori* probabilities.

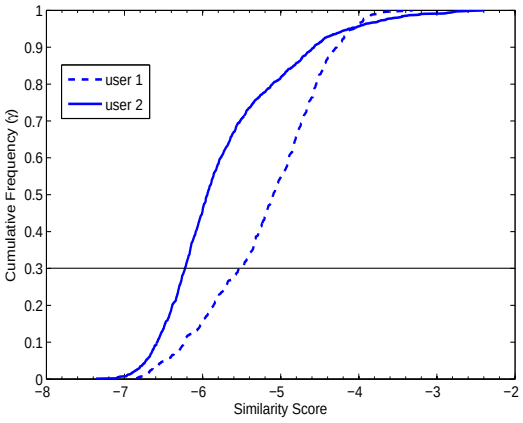


Figure 3. Cumulative histogram of random forgery scores regarding two different users.

IV. Simulation Results

In the first set of experiments, Gentle AdaBoost[3] [11] was used as classifier, and 20 genuine signatures taken from DB_{exp} *versus* 20 random forgeries taken from DB_{dev} where used as training set. Although the maximum number of hypothesis was set to 1000, the algorithm reached training error equal to zero with a single hypothesis per writer (where a hypothesis corresponds to a decision stump, that is, a single-level decision tree constructed from the likelihoods of a single HMM).

The averaged ROC curve representing the proposed system with AdaBoost is indicated by the (+)-dashed line in Fig. 4. The circle-dashed curve corresponds to a traditional feature-based system using HMMs, designed such as described in Introduction. This baseline system uses a single HMM per writer as classifier, where the number of states is selected through the cross-validation procedure described in [3]. Table I (a) and (b) present the error rates on the test data for both systems regarding different operating points (γ), where FPR is calculated regarding three forgery types: random, simple and skilled. As occurred with the validation set (see ROC curves), the DR-based system with AdaBoost provided a significant reduction in the error rates for $\gamma = 1$. For other γ values, both systems performed similarly. Note

[3]Differently from the original AdaBoost algorithm, Gentle AdaBoost deals with overfitting by assigning less importance to outliers.

that the same user-specific thresholds used in the generation of the ROC curves were applied to the test set.

Table I
OVERALL ERROR RATES (%) ON TEST.

(a) Feature-based system

γ	FNR	FPR *(random)*	FPR *(simple)*	FPR *(skilled)*	AER
0.93	0.33	8.50	16.50	72.50	24.46
0.94	0.50	7.67	16.00	70.17	23.58
0.95	0.50	6.83	12.83	68.00	22.04
0.96	0.50	6.00	10.83	64.83	20.54
0.97	0.83	5.67	9.00	60.17	18.92
0.98	1.17	4.00	5.67	52.50	15.83
0.99	2.33	2.67	4.00	42.67	12.92
1	12.67	0.33	1.17	19.83	8.50

(b) DR-based system with AdaBoost

γ	FNR	FPR *(random)*	FPR *(simple)*	FPR *(skilled)*	AER
0.93	3.17	8.33	13.50	72.67	24.42
0.94	3.33	7.00	12.00	69.33	22.92
0.95	3.33	6.00	11.50	67.67	22.12
0.96	3.33	4.83	10.83	64.33	20.83
0.97	3.50	4.00	7.67	58.83	18.50
0.98	3.67	3.17	5.17	52.33	16.08
0.99	4.50	1.67	3.50	41.83	12.87
1	13.67	0	0.50	12.00	6.54

(c) DR-based system with Random Subspaces and SVMs

γ	FNR	FPR *(random)*	FPR *(simple)*	FPR *(skilled)*	AER
0.93	1.50	2.33	7.17	44.50	13.87
0.94	1.50	2.33	7.17	44.00	13.75
0.95	1.83	2.33	6.83	42.83	13.46
0.96	2.00	2.17	6.50	41.33	13.00
0.97	2.33	1.83	4.83	40.50	12.37
0.98	2.83	1.50	4.17	37.17	11.42
0.99	3.83	1.50	2.67	33.50	10.37
1	8.33	0.50	0.50	15.50	6.21

Table II
ERROR RATES (%) PROVIDED BY OTHER OFF-LINE SV SYSTEMS.

Ref.	FNR	FPR *(random)*	FPR *(simple)*	FPR *(skilled)*	AER
[9]	9.83	0	1.00	20.33	7.79
[8]	25.32	3.8	4.48	7.8	10.35
[3]	2.17	1.23	3.17	36.57	7.87

The next experiments consisted of applying the Random Subspaces method [7] to the configuration $|w_1| = |w_2| = 15$. For each writer, 100 random subspaces were generated by changing the models in both w_1 and w_2. Then, each subspace was used to train a SVM with RBF kernel [12], whose parameters $\{c, \gamma\}$ were found through the gridsearch technique with 10-fold cross-validation. Finally, the outputs of each classifier were combined by using the majority vote rule. The averaged ROC curve of this system is indicated by the (*)-dashed curve in Fig. 4, and the corresponding results on test data are presented by Table I (c). Note that the DR-based system with ensemble of SVMs provided a reduction in AER from 2.29%, for $\gamma = 1$, up to 10.59%, for $\gamma = 0.93$ when compared to the feature-based system.

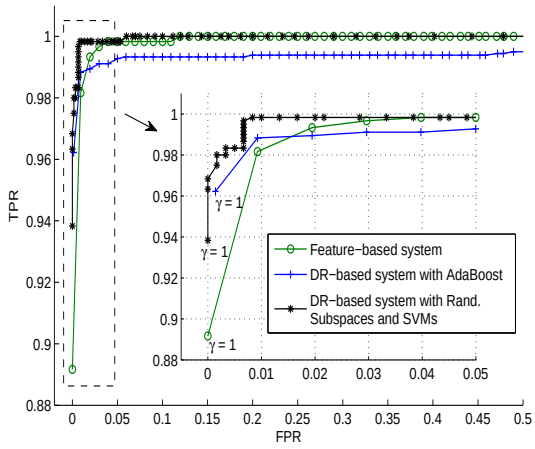


Figure 4. Averaged ROC curves for the baseline and proposed systems. These curves were obtained by using a validation set which contains only genuine signatures and random forgeries.

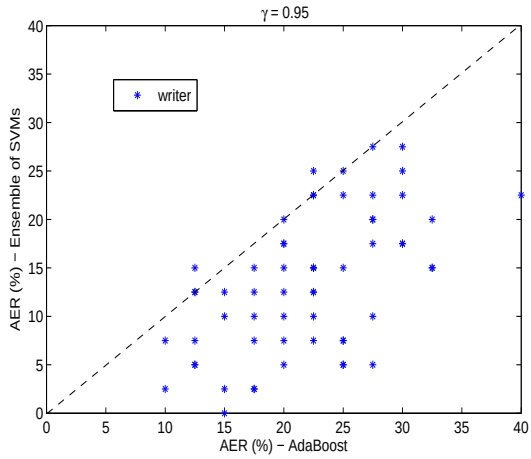
Fig. 5 presents a comparison between both proposed DR-based systems, regarding two different operating points, where the AER on test data was computed invidually for each writer. For γ = 0.95, 85% of the writers obtained improvements with the use of ensembles SVMs. For γ = 1, a slight superiority of ensemble of SVMs over Adaboost was observed. That is, 40% of the writers performed better with ensemble of SVMs, while 33.3% performed better with AdaBoost. For the remaining 26.7%, both systems performed equally.

Finaly, Table II shows the results provided by other systems that use the Brazilian SV database [8] and density of pixels as features. Except in [8], where DR is employed to design a global SVM classifier, the refered articles propose feature-based approaches with user-specific HMMs. By assuming that the objective of these systems is to minimize the AER, we can compare them with $\gamma = 1$ (see last line of Table I (b) and (c)) and conclude that our systems provide the smallest error rates.

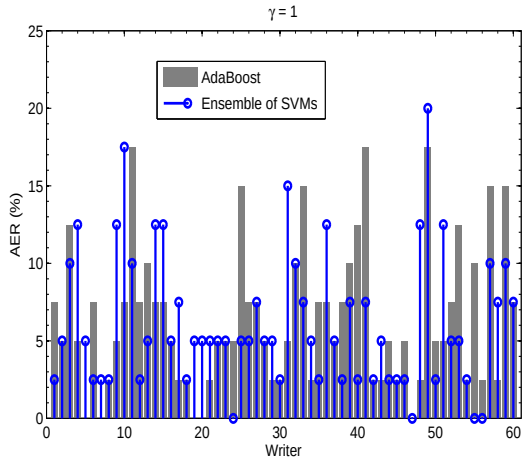
V. CONCLUSIONS

In this paper, a two-stage off-line SV system based on dissimilarity representation [4] was proposed. In the first stage, a set of discrete *left-to-right* HMMs – trained with different number of states and different codebook sizes – was used to measure similarity values that form new feature vectors. In the second stage, these vectors were input to a Gentle AdaBoost classifier or to an ensemble of SVMs in order to obtain the final classification.

During the experiments, the ensemble of SVMs provided better results than AdaBoost in almost all operating points of the ROC space. Since the training set – which contains only true signatures and random forgeries – is easly separable



(a)



(b)

Figure 5. Individual $AERs$ obtained on test for $\gamma = 0.95$ (a) and $\gamma = 1$ (b), using both proposed DR-based systems. In (a), note that writers that had their $AERs$ reduced by using ensemble of SVMs are located below the dashed line.

after the DR process, AdaBoost solves the problem by using one hypothesis (i.e., a single weak classifier). However, in the test phase, other forgery types are considered, and the problem becomes more difficult. The use of ensemble of SVMs therefore provides a robust solution and a higher level of performance on unknown data. When compared to the feature-based system, the proposed DR-based systems provide significant reduction in the error rates. The main reason for this improvement is the fact that DR allows to model both genuine and impostor's subspaces through a set of HMMs, which does not occur with the feature-based system using a single HMM per writer.

The proposed approach may require greater computational complexity than a traditional approach due to the overproduction of candidate HMMs. However, only those HMMs associated to the selected subspaces should be kept in memory.

VI. Acknowledgments

This research has been supported by the Fonds Québécois de la Recherche sur la Nature et les Technologies and by the Natural Sciences and Engineering Research Council of Canada.

References

[1] L. Batista, D. Rivard, R. Sabourin, E. Granger, and P. Maupin, "State of the art in off-line signature verification," in *Pattern Recognition Technologies and Applications: Recent Advances*, 1st ed., B. Verma and M. Blumenstein, Eds. IGI Global, 2007.

[2] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[3] E. Justino, F. Bortolozzi, and R. Sabourin, "Off-line signature verification using hmm for random, simple and skilled forgeries," in *International Conference on Document Analysis and Recognition*, 2001, pp. 105–110.

[4] M. Bicego, V. Murino, and M. Figueiredo, "Similarity-based classification of sequences using hidden markov models," *Pattern Recognition*, vol. 37, no. 12, pp. 2281–2291, 2004.

[5] E. Pekalska, P. Paclik, and R. Duin, "A generalized kernel approach to dissimilarity based classification," *Journal of Machine Learning Research*, vol. 2, p. 2001, 2002.

[6] Y. Freund and R. E. Schapire, "A short introduction to boosting," in *In Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*. Morgan Kaufmann, 1999, pp. 1401–1406.

[7] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, 2001.

[8] D. Bertolini, L. Oliveira, E. Justino, and R. Sabourin, "Reducing forgeries in writer-independent off-line signature verification through ensemble of classifiers," *Pattern Recognition*, vol. 43, pp. 387–396, 2010.

[9] L. Batista, E. Granger, and R. Sabourin, "Improving performance of hmm-based off-line signature verification systems through a multi-hypothesis approach," *Int. Journal on Document Analysis and Recognition*, 2009.

[10] A. Jain and A. Ross, "Learning user-specific parameters in a multibiometric system," in *International Conference on Image Processing (ICIP)*, 2002, pp. 57–60.

[11] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: A statistical view of boosting," *The Annals of Statistics*, vol. 38, no. 2, pp. 337–374, 2000.

[12] C. Chang and C. Lin, "Libsvm: a library for support vector machines," in *http://www.csie.ntu.edu.tw/˜ cjlin/libsvm*, 2001.