Combining Multiple HMMs Using On-line and Off-line Features for Off-line Arabic Handwriting Recognition

Mahdi Hamdani[1] Haikal El Abed[2] Monji Kherallah[1] Adel M. Alimi[1]
[1] Research Group on Intelligent Machines (REGIM), University of Sfax, ENIS, B.P.W, Sfax, 3038, Tunisia
[2] Institute for Communications Technology (IfN), Schleinitzstrasse 22, 38092 Braunschweig, Germany
{mahdi.hamdani, elabed, monji.kherallah, adel.alimi}@ieee.org

Abstract

This paper presents an off-line Arabic Handwriting recognition system based on the selection of different state of the art features and the combination of multiple Hidden Markov Models classifiers. Beside the classical use of the off-line features, we add the use of on-line features and the combination of the developed systems. The designed recognizer is implemented using the HMM-Toolkit. In a first step, we use different features to make the classification and we compare the performance of single classifiers. In a second step, we proceed to the combination of the on-line and the off-line based systems using different combination methods. The system is evaluated using the IFN/ENIT database. The recognition rate is in maximum 63.90% for the individual systems. The combination of the on-line and the off-line systems allows to improve the system accuracy to 81.93% which exceeds the best result of the ICDAR 2005 competition.

1. Introduction

In spite of the great progress shown in Arabic handwriting recognition during the last years, this field of research remains a challenge task. The implementation of Arabic handwriting recognition in real life applications still being limited and need to be improved.

The main purpose of handwriting recognition is to facilitate and make faster the communication with the machine. A recognition system can be based on the spatial data using an image as input (off-line case) or on the temporal data using a list of points representing the trajectory of writing (on-line case). The transformation of the input signal from a representation to another one is possible in the two ways using several techniques.

Combination of classifiers is applied for different applications related to pattern classification. The combination of multiple classifiers is applied on their outputs, and it is based on the degree of support produced by each system. The degree of support consists on the confidences for a given input. The application of this combination is adopted in the aim to deduct a better decision from multiple opinions of different classifiers. Different methods of combination was applied to handwriting recognition and have shown better results than single classifiers [18]. Combination of multiple classifiers was used also in Arabic handwriting recognition [1, 2, 3, 12].

Many authors focused their works on the combination of the on-line and the off-line systems. Two strategies are used to combine these different sources of information. The first one consists of combining them at the feature level. In [4] a method to combine on-line and off-line features is presented. The two streams are synchronized before the step of feature extraction. The second method of combination is applied at the classifier level. The combination is based on the outputs of the off-line and the on-line recognizers [17]. We can say that the combination here is a post-processing step in the recognition process.

Hidden Markov Models (HMMs) [15] are statistical models used in many tasks in pattern recognition and notably in speech recognition. Arabic handwriting recognition benefited from HMMs in the off-line systems and less in the on-line ones [5]. Many advantages are offered by the use of HMMs. We can cite the implicit segmentation of the Arabic handwriting which could be very difficult in the explicit case. The HMMs are also used for their resistance to noise. The use of HMMs on any recognition task requires a temporal information of the input signal. For the use of the HMMs in the off-line handwriting recognition, we generally use a sliding window in the direction of the writing (right to left in the Arabic language) to allow the segmentation and the recognition.

Actually, the off-line systems in handwriting recognition use only the spatial data. This problem can be solved by the use of the temporal information of the writing. We present

978-0-7695-3725-2/09 $25.00 © 2009 IEEE
DOI 10.1109/ICDAR.2009.40

in this paper an off-line handwriting recognition system based on the combination of multiple HMMs. These HMMs are based on different features. The used features are chosen from the state of the art in the domain. The basic idea is to combine an on-line recognizer with off-line ones trying so to overcome the limits of the off-line systems.

This paper is organized as follows. In section 2 the different features are briefly presented. In section 3 we present the designed recognition system with the use of HTK followed by the tests and results of this work in section 4. Finally, we present some conclusions and future works.

2. Feature Extraction

The step of the feature extraction prepares the data to be used by a classifier. In our case, we use HMMs which require a temporal information of the input data. This information is unavailable in images to be recognized, so we have to generate it. We use the sliding window technique as described in [16] for the extraction of the off-line features. This window scans the input image from right to left in the direction of the Arabic writing. The size of the window and the overlap between consecutive windows are system parameters. The values of these parameters are chosen after multiple tests. The technique of the sliding window is presented in Figure 1. This technique of feature extraction is typically used in the HMM based systems for the recognition of off-line text. The sliding window is used by the systems with the higher accuracy in the ICDAR 2007 competition. We use this technique for the different off-line features.

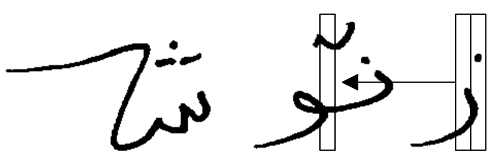

Figure 1. Sliding window

We use three off-line methods of feature extraction and one on-line method. The choice of these features is based on the study of the state of the art. The different features are used successfully in different applications in handwriting recognition.

2.1. Pixel Values (OFF-1)

The pixel values are used in [14] with the application of a normalization method of the Arabic handwriting. We retrieve these values from normalized images and apply to them a Karhunen Loeve Transformation (KLT) to reduce the number of features used by the system. We proceeded to multiple tests with the variation of the size of the feature vector from 27 to 150 values chosen from the initial vector composed by 360 values. We concluded that the best results are given by the use of 150 values. These features are computed from a window with a size of 5 pixels and 3 pixels as overlap between consecutive windows.

2.2. Densities and Moment Invariants (OFF-2)

These features were used in [10] for the recognition of handwritten digits. The density of black pixels is calculated by a re-sampling procedure, the window is divided into cells and the value of the density of black pixels of each cell is used. The re-sampling procedure is presented in Figure 3. The Moment Invariants of Hu [11] are calculated for each window and concatenated with the density values. The feature vector is so composed of 92 values computed from an 8 pixel window width. An overlap is not used for these tests.

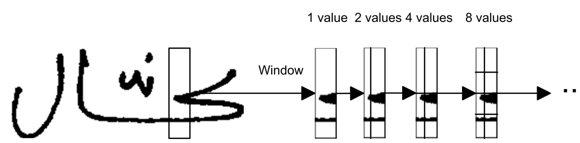

Figure 2. Calculating densities by the re-sampling procedure

2.3. Pixel Distribution and Concavities (OFF-3)

These features were used in [8] for the recognition of Arabic handwritten words. We can say that these features are the combination of two different strategies in the feature extraction step. The pixel distribution features are considered as purely statistical features, but the concavities are structural features. We use 16 features related to the pixels distribution and 8 features representing the different possibilities of concavities extracted from a sliding window with a size of 8 pixels and without overlap.

2.4. On-line Features (ON-1)

In On-line handwriting recognition we can benefit from the use of the spatio-temporal information in the feature extraction step. The use of on-line features for off-line handwriting recognition is possible since the availability of methods for the recovery of the temporal information from the off-line trace. We use the technique described in [9]. This technique allows to have the on-line trace of the writing given an image as input.

The used on-line features are introduced in [6]. We use the extracted features to model the on-line handwriting on the basis of the beta-elliptic approach. After the transformation of the input image into a sequence of coordinates, each part of word (PAW) is segmented into a sequence of graphemes. After that, 21 features related to the beta-elliptic modelling are extracted from the grapheme.

3. HTK based HMM Recognizer

The recognizer is based on the combination of multiple HMMs and implemented using the HMM Toolkit. This section contains a brief description of the HTK engine followed by the presentation of the HMM architecture. Finally, we present the used combination methods.

3.1. HTK Engine

After the feature extraction, the object is to run the trained HMM recognizer. We have used the HMM Toolkit (HTK) for the implementation of the system [19]. HTK is a portable toolkit for HMM based recognizers. This toolkit is developed initially for speech recognition. HTK supports multiple steps in the recognition process: data preparation (or feature extraction), training, recognition and post-processing. The data preparation supports only the speech data, so we do not use HTK for this step. For the other steps, we can adapt HTK to support our data. A HTK system have to be prepared for each feature extraction method. The recognition process using the HTK engine is presented in Figure 3. The features are used to train the HMM to

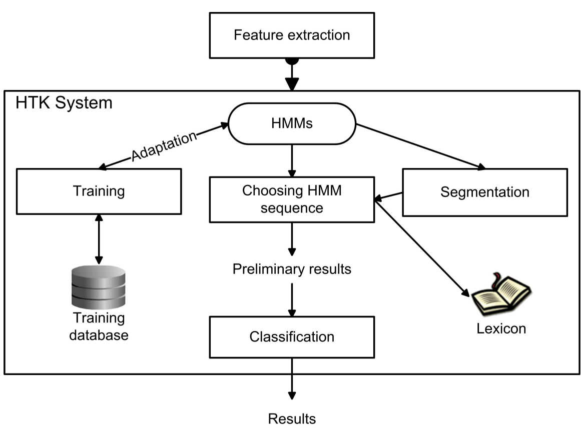

Figure 3. The recognition process in the HTK engine

have the different models of the Arabic characters. The decoding allows the segmentation and the recognition of the input words.

3.2. HMM Architecture

In this section, we present the designed HMM architecture using the HTK engine presented in the previous section. We use left to right discrete probability HMMs for the modelling of Arabic characters. Basically, the Arabic language contains 28 letters. However, in our system there are more than 160 HMM models. This is due to the existence of multiple shapes for the same letter and multiple ligatures used by the writers.

There are two stages in the training process. The first stage corresponds to the construction of the initial codebook and the second one is the re-estimation of the HMM parameters. We use a codebook with 256 entries for all the features. We use the Baum-Welch re-estimation algorithm.

The statistical decoding is performed using the Viterbi algorithm. We use the beam search to speed up the recognition. A dictionary is also used in the decoding. A quantization of the feature vector using the constructed codebook is required before the decoding process. As a result, we associate for an input image a list including the most probable labels and the corresponding confidences.

3.3. Classifiers Combination

We have presented in section 2 the different implemented features. Each HMM is trained and optimized using one of these feature extraction methods. The idea to combine these different systems is based on the assumption that the combination of multiple classifiers gives better results than single ones. The combination of the on-line and the off-line systems was successful for the previous works. We test this combination using the method introduced in [7]. The combination is applied at the classifier level using the confidences produced by the different systems. These confidences are normalized before the combination of the systems decisions. After that, several combination schemes are tested. These different schemes are based on logical rules that are presented in what follows.

3.3.1 Weighted Majority Voting (W_{Mv})

The idea of the majority voting is relatively simple, the reliability of a result is calculated on the basis of the number of recognizers that decide it. The simple majority voting is based only on the number of the recognizers and their relative decisions. This method does not put in evidence the confidence of each classifier for its decision. The weighted majority voting (W_{Mv}) allows the consideration of the confidences in the combination. The equation 1 shows how to calculate W_{Mv} for n systems.

$$W_{Mv}(x_k) = \max_{S_{l,1} \in \{S_{1,1},\dots,S_{n,1}\}} (\sum_i w_{i,1}(x_k)) \qquad (1)$$

3.3.2 Borda (B_c)

The Borda combination method uses the ranking information of the results (r-best results) provided by the different classifiers to decide. It returns all the possible results in a complete ranked list. The calculation of the Borda count (B_c) is presented in the equation 2.

$$B_c(x_k) = \max_{S_{l,j}, w_{l,j}} (\sum_{i=1}^{n} (r - rank(S_{i,j}(x_k), w_{i,j}(x_k)) + 1)) \quad (2)$$

3.3.3 Rank Count (R_c)

The rank count method is based on the attribution of a cost function c_i for each classifier. To use this method we have to define also a system confidence value as a rank function for the different classifiers. The rank count is presented in the equation 3.

$$R_c(x_k) = \max_{S_{l,j}, w_{l,j}} (\sum_{i=1}^{n} (a_i - c_i(rank(S_{i,j}(x_k), w_{i,j}(x_k))))) \quad (3)$$

As it was the case in [7], we use two forms of this method. In the first one (R_c), the weight for each class is assigned as output of the cost function c_i. In the second one, the output of the cost function c_i is the product of the rank and the weight of each word image x_k (M_{Rc}: Modified Rank Count). The classifier confidence a_i is equal to zero in both forms of the method.

4. Experimental results

4.1. Data

The IFN/ENIT database version 2.0 with patch level 1e (v2.0p1e) [13] is used for the tests of the system. This database contains 32492 images of Arabic handwritten words (Tunisian town/village names). The database is divided in 5 sets (a-e) with an equitable distribution in the number of examples.

4.2. Tests setup

All the systems are trained and tested using the data described in section 4.1. The codebook size is 256 for all the systems. We use right to left discrete HMMs. A recapitulation of the different parameters used in the tests are presented in Table 1.

4.3. Results

The training is performed using the sets a, b, c and d. We test the system with the sets d and e with the respect of the conditions of the ICDAR 2005 competition [13]. In fact, we did not have the possibility to test the system with same conditions of the ICDAR 2007 competition because of the unavailability of the sets f and s. The recognition rates for each system are given in Table 2. We can see that the system OFF-1 has the best results for this test. However, the system OFF-3 has not shown good results in spite of the good accuracy of the original system presented in [8]. This fact can be explained by the difference between the original system and this one in the HMM architecture and the system design.

Table 1. Recapitulation of the systems parameters

System	Feature vector size	Number of states
OFF-1	150	5
OFF-2	92	3
OFF-3	24	3
ON-1	20	3

Table 2. Recognition rates in % corresponding to the individual systems

Systems	set d	set e
Pixel values (OFF-1)	86.28	63.90
Densities + Moment invariants (OFF-2)	83.86	51.57
Pixel distribution + Concavities (OFF-3)	67.68	49.48
On-line Features (ON-1)	81.21	50.01

Each system gives its r-best results for all the input images with the corresponding confidences. After that, These results are used to apply the different methods of combination as described in section 3.3. The results of the combination of the systems are presented in Table 3 and Table 4 respectively for sets d and e.

Table 3. Recognition rates in % corresponding to the combination methods for the set d

Systems	OR	W_{Mv}	B_c	R_c	M_{Rc}
Best 2 systems	94.94	87.75	91.54	93.20	92.75
Best 3 systems	97.62	93.13	95.93	96.11	96.78
4 systems	98.00	94.34	95.92	95.90	**96.97**

The use of the operator OR presents the perfect combination scheme that can be reached with these systems knowing the ground truth information. As we can see, the M_{Rc} method gives the best results with a recognition rate of 96.97% for the set d and 81.93% for the set e. This result

Table 4. Recognition rates in % corresponding to the combination methods for the set e

Systems	OR	W_{Mv}	B_c	R_c	M_{Rc}
Best 2 systems	76.83	64.86	69.04	70.20	70.38
Best 3 systems	84.19	71.03	76.30	74.99	77.62
4 systems	87.57	75.20	80.72	78.37	**81.93**

allows to exceed the best results found in the ICDAR 2005 competition which was 75.93% for the test set. The combination of all the off-line systems gives only the recognition rate of 76.96%. These results show that the combination of the on-line and the off-line systems is beneficial for the progress of Arabic handwriting recognition.

5. Conclusions and future works

In this paper, we present an off-line handwriting recognition system based on the combination of multiple HMMs. The different HMMs are based on on-line and off-line features. We demonstrate that the combination of an on-line system with off-line ones gives better results than the combination of multiple off-line recognizers. The combination is applied at the classifier level using the confidence scores. The combination of the on-line and the off-line systems can be also applied at the feature level, but we have to synchronize these different sources of information.

Acknowledgements.

This work was realized within the framework of the DAAD project "In the way of information society". We are grateful for Dr. Volker Märgner for his helpful and precious advices. In addition, we acknowledge the financial support of this work by grants from the General Direction of Scientific Research and Technological Renovation (DGRST), Tunisia, under the ARUB program 01/UR/11/02.

References

[1] A. M. Alimi. An Evolutionary Neuro-Fuzzy Approach to Recognize On-Line Arabic Handwriting. In *Proc. of the* 4^{th} *Inter. Conf. on Document Analysis and Recognition (ICDAR)*, pages 382–386, 1997.

[2] A. M. Alimi. A neuro-fuzzy approach to recognize Arabic handwritten characters. In *Inter. Conf. on Neural Networks*, volume 3, pages 1397–1400, 1997.

[3] A. M. Alimi. Evolutionary computation for the recognition of on-line cursive handwriting. *IETE journal of research*, 48(5):385–396, 2002.

[4] T. Artieres, N. Gauthier, and P. Gallinari. A Hidden Markov Models combination framework for handwriting recognition. *Inter. Journal on Document Analysis and Recognition*, 5:233–243, 2002.

[5] N. Ben Amara, A. Belaïd, and N. Ellouze. Utilisation des modèles markoviens en reconnaissance de l'écriture arabe : Etat de l'art. In *Colloque Inter. Francophone sur l'Ecrit et le Document (CIFED)*, pages 181–191, 2000.

[6] H. Boubaker, M. Kherallah, and A. M. Alimi. New strategy for the on line handwriting modeling. In *Proc. of the* 9^{th} *Inter. Conf. on Document Analysis and Recognition (ICDAR)*, volume 2, pages 23–26, 2007.

[7] H. El Abed and V. Märgner. Improvement of Arabic handwriting recognition systems - combination and/or reject? In *Document Recognition and Retrieval XVI, IS&T/SPIE Symposium on Electronic Imaging*, number 7247–10, 2009.

[8] R. El-Hajj, L. Likforman-Sulem, and C. Mokbel. Arabic Handwriting Recognition Using Baseline Dependant Features and Hidden Markov Modeling. In *Proc. of the* 8^{th} *Inter. Conf. on Document Analysis and Recognition (ICDAR)*, volume 2, pages 893–897, 2005.

[9] A. Elbaati, M. Kherallah, A. Ennaji, and A. M. Alimi. Temporal Order Recovery of the Scanned Handwriting. In 10^{th} *Inter. Conf. on Document Analysis and Recognition (ICDAR)*, 2009. Accepted.

[10] T. M. Hamdani and A. M. Alimi. *Artificial Neural Networks and Genetic Algorithms*, chapter β-SVM a new support vector machine, pages 63–68. Springer, 2003.

[11] M.-K. Hu. Visual Pattern Recognition by Moment Invariants. *IRE Transactions on Information Theory*, 8(2):179–187, 1962.

[12] M. Kherallah, L. Haddad, A. M. Alimi, and A. Mitiche. On-line handwritten digit recognition based on trajectory and velocity modeling. *Pattern Recognition Letters*, 29:580–594, 2008.

[13] V. Märgner, M. Pechwitz, and H. El Abed. ICDAR 2005 Arabic Handwriting Recognition Competition. In *Proc. of the* 8^{th} *Inter. Conf. on Document Analysis and Recognition (ICDAR)*, volume 1, pages 70–74, 2005.

[14] M. Pechwitz and V. Märgner. HMM Based Approach for Handwritten Arabic Word Recognition Using the IFN/ENIT - Database. In *Proc. of the* 7^{th} *Inter. Conf. on Document Analysis and Recognition (ICDAR)*, volume 2, pages 890–894, 2003.

[15] L. R. Rabiner. A Tutorial on hidden Markov models and selected applications in speech recognition. In *Proc. of the IEEE*, volume 77, pages 257–286, 1989.

[16] R. Schwartz, C. LaPre, J. Makhoul, C. Raphael, and Y. Zhao. Language-independent OCR using a continuous speech recognition . In *Proc. of the* 13^{th} *Inter. Conf. on Pattern Recognition (ICPR)*, volume 3, pages 99–103, 1996.

[17] R. Seiler, M. Schenkel, and F. Eggimann. Off-Line Cursive Handwriting Recognition Compared with On-Line Recognition. In *Proc. of the* 13^{th} *Inter. Conf. on Pattern Recognition (ICPR)*, volume 4, pages 505–509, 1996.

[18] L. Xu, A. Kryzak, and C. Y. Suen. Methods of Combining Multiple Classifiers and Their Applications to Handwriting Recognition. *IEEE Trans. Syst., Man, Cybern.*, 22(3):418–435, 1992.

[19] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. A. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland. *The HTK Book*. Cambridge University, 2006. for HTK Version 3.4.