International Journal of Soft Computing 2 (1): 5-10, 2007

Arabic Handwriting Word Recognition Based on a Hybrid HMM/ANN Approach

Narima Zermi, Messaoud Ramdani and Mouldi Bedda
Department of Electronic, Faculty of Engineering, Badji Mokhtar University, Annaba,
Laboratoire d'Automatique et des Signaux d'Annaba (LASA)
P.O.Box 12, Annaba-23000, Algeria

Abstract: This study describes a hidden Markov model using a grapheme neural networks approach designed to recognize off-line unconstrained Arabic handwritten words. After pre-processing, a word image is segmented into characters or pseudo-characters called graphemes and represented by a sequence of observations. Each observation consists of a set of global and local features that reflect the geometrical and topological properties of a grapheme accompanied with information concerning its affiliation to one of five predefined groups. Within its group, the classification of a grapheme is done by a neural network trained with fuzzy class memberships rather than crisp class memberships as desired outputs because it results in more useful grapheme recognition modules for handwritten word recognition. The experimental results on a test database are presented to demonstrate the reliability of this study.

Key words: Arabic handwritten word recognition, grapheme, neural networks, hidden markov models

INTRODUCTION

The automatic handwriting recognition is not an easy task and despite the impressive progress achieved, the results are still far from human performances; hence, this subject continue to be a challenging task associated with a panoply of difficulties due mainly to the great amount of variability and uncertainty present in handwritten texts. According to the way handwriting, data are acquired, on-line and off-line recognition systems can be distinguished. In the former category, the data is a sequence of pixels drawn by the user on a digitized table and transmitted to the system during the writing in a dynamic way, while in the latter category, the data provided to the system is a static representation obtained with a scanning device after the writing is completed[1-3].

Many research studies in the area have revealed that off-line handwriting recognition is a more difficult task, because of the temporal order and the dynamic information, such as the number and the order of the strokes is not available as in on-line case. Motivated by this observation, several studies have attempted to automatically reconstruct the temporal order of off-line signals. In general, these studyes are based on a number of heuristics and local analysis of handwriting signal. The heuristics assume certain preference for the direction of the writing process. In the remainder of this study, we limit ourselves to off-line Arabic handwriting recognition by adopting a specific modelling framework based on the concept of graphemes to tackle the problem of cursive nature in Arabic scripts. This study capitalizes on the skeleton graph of a word image which consists of a number of links where each link starts and terminates with a feature point. In this way, the skeleton of the word is transformed into a sequence of feature vectors, which allows the recognition by the analytical techniques.

In this study, advanced computer recognition systems for handwritten characters generally have non difficulties in recognizing samples, which are well formed and closely resemble the prototypes for each class. However, the fact that many pseudo-characters (pieces or union of characters) can look like characters, results in a very high potential for false matches. In this study, we should note that handwritten word recognition by computer consists of more than just isolating and reading the individual characters in a word. In common with most visual pattern recognition applications, the use of contextual information to read words is able to enhance the performance. Many alphabetic characters are ambiguous when read out of context. Some examples of ambiguity are shown in Fig. 1.

Corresponding Author: Narima Zermi, Department of Electronic, Faculty of Engineering, Badji Mokhtar University, Annaba, Laboratoire d'Automatique et des Signaux d'Annaba (LASA) P.O.Box 12, Annaba-23000, Algeria

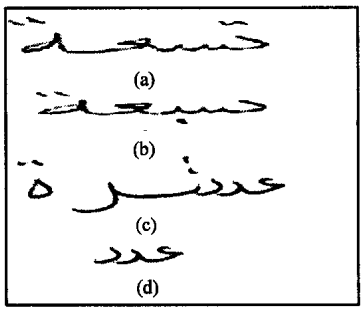


Fig. 1: Ambiguous patterns in handwritten words

Table 1: Comparison of various scripts

Characteristics	Arabic	Latin	Hebrew	Hindi
Justification	R-to-L	L-to-R	R-to-L	R-to-L
Cursive	Yes	No	No	Yes
Diacritics	Yes	No	No	Yes
Number of Vowels	2	5	11	-
Letters shapes	1-4	2	1	1
Number of letters	28	26	22	40
Complementary characters	3	-	-	-

From all the previous considerations, one can affirm that designing an efficient segmentation-based word recognition system is a difficult task. The authors of[4] have proposed a system that supports the use of fuzzy characters class membership in assigning characters confidence for word recognition. Another lexicon-based, handwritten word recognition module is proposed in[2]. This system combines a segmentation-free and segmentation-based techniques in order to improve the performance of the overall system. A segmentation-free technique is proposed in[5] to improve the performance when the handwritten words are not easily segmented. The technique uses a robust statistical multiresolution analysis of the word profiles. This study uses a recent framework called hidden wavelet tree (HWT)[6]. The observation sequences used by the models do not require segmentation of a word image into characters. Thus, this study has advantages over segmentation-based approaches. Therefore, we hope to develop a combination scheme to fuse the two word recognition strategies to complement each other and to improve the overall classification performance.

The characteristics of an arabic text: The most obvious characteristics of the Arabic language is that Arabic scripts are inherently cursive. Arabic script, which comprises 28 main characters and which is written from right to left, is used by many nations: Arabs, Curds, Persians and Urdus. The shape of the letters is context sensitive, depending on their position within a word (beginning, middle, end, Isolated). In Arabic, the letters represent only consonants or long vowels. Short vowels are represented by 14 optional diactrical marks written over or under the letters. Table 1 outlines a comparison of the various characteristics of Arabic, Latin, Ubrew and Hindi Scripts[7-11].

From a text to grapheme: The recognition system can be divided into three main phases: pre-processing and segmentation of a word into strokes called graphemes, classification of graphemes and word recognition. In the following subsection, we describe the steps to obtain grapheme.

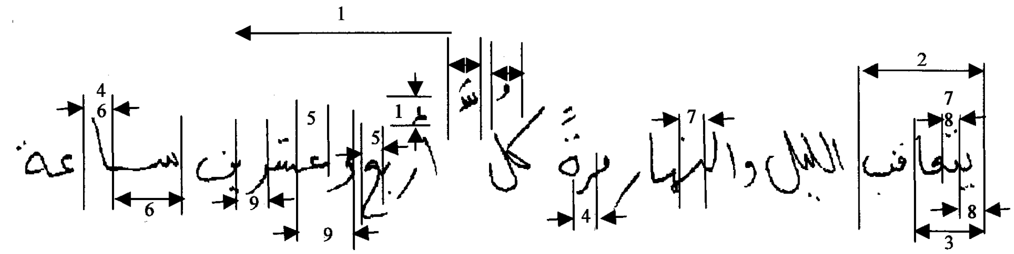
Pre-processing and segmentation: In the first step, the extraction of handwritten data from noisy grey-level images is done. This is achieved by an effective denoising or noise removal procedure followed by a thresholding and a series of horizontal and vertical projections to detect each word into a rectangular window. Then the binary image of the word to be recognized is thinned[12,13]. Then the detection of the base line indicated by the line of writing in the zone of the word is carried out. To obtain the transition profile, the number of black to white transitions and vice-versa Fig. 2 is counted within a certain band of the image for each horizontal line. The follow-up of partial contour which consists in tracing an upward contour and a downward contour of the text line allows a good localization of the lines and then of the words[13].

Segmentation in graphemes: To avoid the difficult task of segmenting a cursive word into characters, we have preferred to model the handwriting with pseudo-characters that we call graphemes. However, the number of those graphemes would be small as possible in order to make their combination simple and effective.

Therefore, a grapheme is defined as a binary sub image or equivalently as a continuous curve represented by a string of coordinates (Xi, Yi), where the order of these coordinates can be assimilated as an approximation of the pen movement during writing. The definition of feature points (cross, branch points and line ends) is given in Fig. 3.

The idea of the grapheme finder algorithm is to find the start point using the algorithm given in Fig. 5, then to find the end point and finally to trace the grapheme from the start to the end. Hence, the grapheme is isolated and sorted from the thinned image and its binary image is also obtained[3].

Extraction of the start point: the extraction of the stroke-finder algorithm is to find the start point using the algorithm given in Fig.5. A search for a black point is done from right to left.



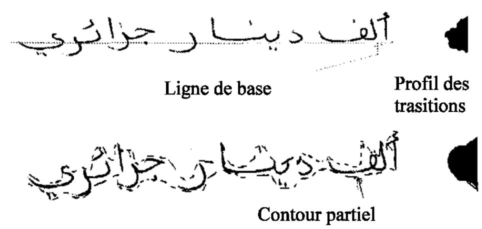


Fig. 2: Example of the transition

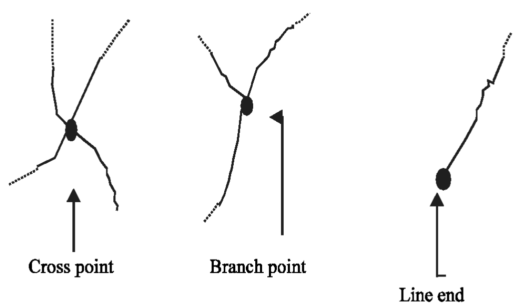


Fig. 3 Feature points

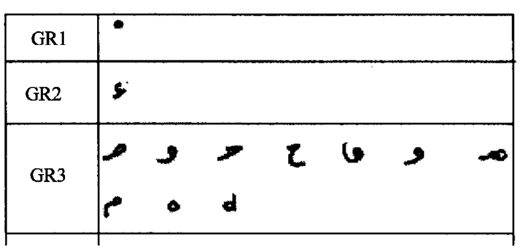


Fig. 4: The different segments that can be generated by the segmentation module

Extraction of the end point: The point that represents the current stroke are followed until an end point is found. An end point can be one of the feature point (a branch point, a cross point or a line point end).

Various groups of graphemes: The different graphemes generated by the segmentation module fall into five groups:

Group1 (Gr1): points defined as isolated segments with a very small length.

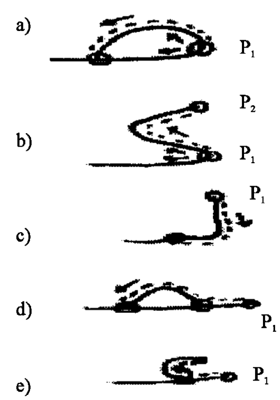


Fig. 5: Extraction of starting point

Group2 (Gr2): Hamza defined as a small isolated Zig-Zag.

Group3 (Gr3): Segments with loops Fig. 4.

Group4 (Gr4): Segments without loops and which finish with connections or intersections.

Group5 (Gr5): Segments without loops and which finish with end point.

The recognition process: There are two main approaches to tackle the difficulties associated with the cursive nature of Arabic scripts: The holistic study and the analytical study. The holistic study treats the word globally by extracting some features from without any segmentation. The analytic study decomposes the word into smaller units. In the present study, the recognition process falls in the second category. It is based on hidden Markov models that uses feed-forward neural networks to classify the different graphemes. In this way, better estimates of the observation probability distribution become possible. The following subsections describe the grapheme recognition with neural networks and then the hidden Markov models.

Grapheme recognition

Neural network structure: The task of the feed-forward neural networks is to classify the whole set of different

shapes of grapheme. A single neural network was tried to do the whole classification task but resulted in a weak performance. This is due to the resemblance between many graphemes. In order to improve the classification performances, separate networks were used for the groups Gr3, GR4 and Gr5.

The networks were trained using back propagation algorithm. Each has input, output and two hidden layers. Each hidden and output unit has a bias. Each network has 120 input units, 65 units in the first hidden layer, 30 in the second hidden layer and M output units (M=10 for Gr3, M=13 for Gr4 and M=11 for Gr5).

Computation of desired outputs: The desired outputs were set using the fuzzy-K-Nearest neighbour algorithm proposed by Kell *et al*[15]. We chose to use 15 nearest neighbours of a training set using Euclidean distance and to assign membership values were used as target outputs to train the neural networks.

Hidden markov models: Hidden Markov Models (HMMs) are widely used in the field of pattern recognition. Their original application was in speech recognition. Because of the similarities between speech and cursive handwriting recognition, HMMs have become very popular in handwriting recognition as well.

A good introduction to hidden Markov models can be found in the study[15]. HMM have become a very popular technique for cursive recognition, for on-line as well as off-line systems. The basic idea is to model cursive writing as a hidden underlying structure, which is not observable. This underlying structure is a canonical representation of the reference patterns (e.g. a sequence of letters).

Deformations in their drawing produce observable symbols. Hence, HMMs are able to model the fact that for example different letters may have similar shapes and that a given letter may be written in many different ways.

A discrete HMM (DHMM) is entirely defined by a set of states S a fixed vocabulary of symbols V, probabilities of initial states π, transition probabilities between states A and observation probabilities of symbols within states B.

A fully described HMM is characterized by 4 elements:

- N, the number of states, individual states are denoted $S=\{s_1,s_2,\ldots\ldots s_N\}$ and the state at time t: qt
- M, the number of possible observation symbols, individual observations are denoted $V=\{v_1,v_2,\ldots\ldots v_M\}$;
- $\pi=\{\pi_i\}$, the initial state distribution.
- $A = \{q_{ij}\}$, the state transition probability distribution where

$$a_{ij} = P\left[q_{t+1} = S_j \;/\; q_t = S_i\right] \qquad 1 \le i, j \le N$$

- B={bj(k)}, the observation symbol probability distribution in state j, where

$$b_j(k) = P\left[V_k \text{ at } t \;/\; q_t = S_j\right] \quad 1 \le j \le N, \quad 1 \le K \le M$$

RESULTS

Collection of the data: The data of this project are extracted from a database of handwritten Arabic words collected at the department of Electronics; university of Annaba. The lexicon is made essentially of the literal numerals used for checks (48 classes).

Sample of about 5000 Arabic words for the lexicon used in check filing were gathered and stored in separated files. The database was segmented into 2500 training samples were used for the training and 2500 samples for the performance evaluation.

The graphemes base was extracted from the words base to form 1755 samples for the training set (50 by classes for Gr3, Gr4 and Gr5) and a test set of the same size.

Pre-processing: The strategy of smoothing adopted consists of filtering the image on gray level by mathematical morphology; an operation of opening by a square structuring element of size 3×3 pixels. The binarization operation rests upon the calculation of the histogram moments for the gray levels to select an optimal threshold. Then, the localization of the graphemes in rectangular windows permits to characterize them by feature vectors and to classify the word with one of the three categories depending on the number of the grapheme or equivalently the length of the word. In this way, the computation time is reduced because only a subset of candidate words is involved in the recognition process.

Word category	Numbers grapheme
Weak	≤ 4
Average	5-9
Large	≥ 10

Neural networks results: In order to show and illustrate the various stages of processing a handwritten word, an illustrative example is given in Fig. 6.

As far as graphemes of groups (GR3, GR4 and GR5) are concerned, the results of training are shown in

Table 2: Recognition rates of the grapheme neural network for two different feature vectors

	Rate of recognition (bar features)	Rate of recognition (profile of contour)
GR3	89.8%	94.7%
GR4	88.9%	87.9%
Gr5	90.5%	96.5%

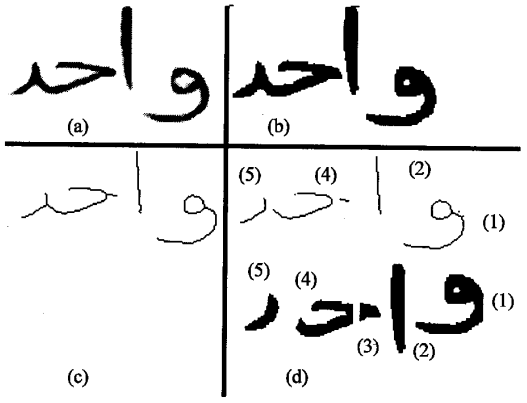


Fig. 6: An example illustrating the various stages of treatment (a) A gray scale image (b) The smoothed binary image (c) The thinned image (d) five segments to be classified by the neural networks based on their features.

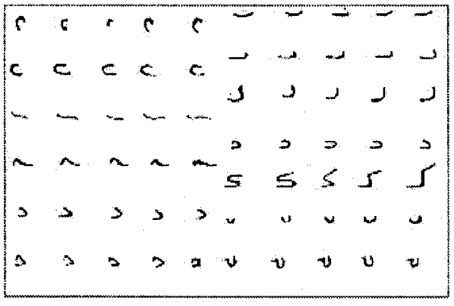


Fig. 7: Some grapheme training samples

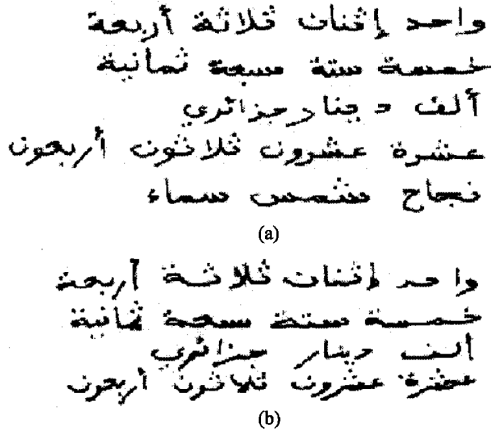


Fig. 8: (a) Some correctly classified word image (b) Some incorrectly classified word images

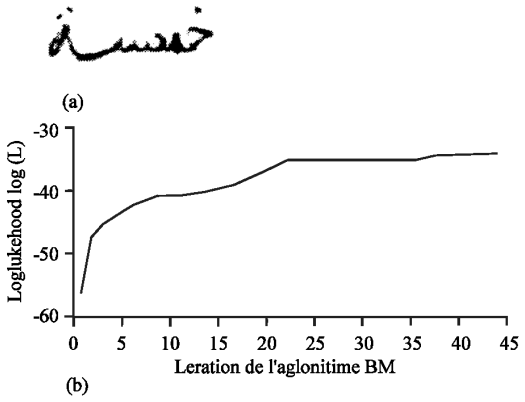


Fig. 9: (a) A sample of the word "five" in Arabic (b) Evolution of logarithm likelihood during the HMM model training for the class "five"

Table 2. This table gives the percentage of the graphemes for which the largest output is associated with the true class (the rate of recognition), which is a standard performance measure of the classifiers.

The HMM used is of continuous probability type with 12 states, for which each grapheme can belong to any state according to a certain probability. If each grapheme is characterized by contour profiles, the rates of recognition on the basis of the previous data are practically about 95%.

CONCLUSION

This study outlines a methodology of combining hidden Markov models and neural networks for Arabic handwriting recognition. The introduction of the graphemes notion is of great importance since it allows to employ a process of segmentation without constraint and subsequently to provide a list of candidate words (according to a length criterion). In order to improve the performances of the system, the fusion of several classifiers at the grapheme level is being implemented and the practical results are very promising. The selection of the most important features in a systematic is as also expected to reduce the complexity of the overall system. The proposed system has been used in legal amount recognition with an acceptable recognition rates and the additional information provided by the digital amount of the bank check is able to improve the recognition accuracy.

REFERENCES

1. Lazzerini, B. and F. Marcelloni, 2001. A fuzzy Approach to 2-D Shape recognition IEEE Transaction on fuzzy systems, pp: 512-516.

2. Madavanath, S. and V. Govindarary, 2005. The role of holistic paradigms in handwritten word recognition IEEE transactions on pattern analysis and machine intelligence, pp: 149-164.
3. Ye Xiangyum and M. Cheriet, 2004. stroke-model-based character extraction from gray level document image IEEE Trans on image processing, pp: 1152-1161.
4. Gader, P.G., M. Mohamed and J. Chang, 1995. Comparison of Crisp and fuzzy character networks in handwritten word recognition, IEEE Trans, Fuzzy Systems, pp: 357-364.
5. Magdi, M. and P. Gader, 1996. Handwritten word recognition using Segmentation-free Hidden Markov Modelling and segmentation-based Dynamic programming Techniques, IEEE PAMI, pp: 548-554.
6. Crouse, M.S., R.D Nowak and R. G. Baraniuk, 1998. Wavelet-based statistical signal processing using hidden Markov m odels, IEEE Trans. Signal Proc., pp: 886-902.
7. Amin, A., A. Kaced, J.P. Hatn and R. Mohr, 1980. Handwritten Arabic Characters recognition by the IRAC system, 5th Int Conf Pattern Recognition, Miami, USA, pp: 729-731.
8. Amin, A., H. AlSadoun and S. Fisher, 1996, Hand-printed Arabic Character recognition system using an artificial networks Pattern Recognition, pp: 663-375.
9. Touj, S., N. Ben Amara and H. Amiri, 2002. Global feature extraction of off-line Arabic handwriting, GEI'2002, Hammamet, Tunisie, pp: 120-126.
10. Snoussi, S. and H. Amiri, 2002. Combination of local and global vision modelling for Arabic Handwritten words recognition Proceddening of the eighth international workshops on frontiers in handwriting recognition (IWFHR'02), , IEEE, pp: 156-164.
11. Sabri A. Mahmoud, 1994, Arabic Character Recognition using Fourier Descriptors and Character Contour Encoding Pattern Recognition, pp: 815-824.
12. Zidouri, A., M. Sarfras and M. Jafri, 2005. Adaptive dissection based sub word segmentation of printed Arabic text IEICE Trans. Information and system, pp: 1649-1655.
13. Zermi, N., M. Ramdani and M. Bedda, 2003. Neuro Markovian hybrid system for handwritten Arabic word recognition. 10th IEEE International Conference on Electronics, Circuits and Systems-ICECS University of Sharjah., united Arabic Emirates.
14. Keller, J. and M.R. Gray, 1985, A fuzzy K-nearest neighbours algorithms IEEE SMC, pp: 580-581.
15. Rabiner, L.R., 1989. A tutorial on Hidden Markov Models and selected applications in speech recognition, Proceeding of the IEEE, pp: 257-285.