HMM-BASED ARABIC HANDWRITTEN CURSIVE RECOGNITION SYSTEM

Ramy Al-Hajj\textsuperscript{1,2) } and Chafic Mokbel\textsuperscript{1) }

\textsuperscript{1) }University of Balamand, P.O. Box 100, Tripoli, Lebanon.
\textsuperscript{2) }Université de Reims, département Mathématiques, Reims, France.

ABSTRACT

In this paper we present the first results of a language-independent handwritten recognition baseline system developed to recognize cursive handwritten words. The system is based on a stochastic Hidden Markov Model (HMM) approach. A continuous density HMM is used in the classification module. Preprocessing and features extraction steps are described in this paper as well as the HMM-based classifier. The features used are based on the distributions of pixels in a sliding window. We have stressed the importance of the detection of a baseline in the image. Some experiments were performed on a benchmark database IFN/ENIT of multi-scripters Arabic handwritten words (Tunisian country/village names) taken from more than 400 writers. The recognition rates obtained are promising and show that the use of a baseline reduces the error rate by 23.5%.

1. INTRODUCTION

The off-line handwritten recognition is one of the challenging problems in the pattern recognition domain. Several approaches have been proposed in this field attempting to reach the human behavior \cite{1}\cite{3}\cite{9}\cite{10}. Currently, the recognition of printed and hand-printed characters is performed with satisfactory accuracy that exceeds 90\% \cite{8}. Whereas, the recognition of cursive handwritten still remains an open problem due to the existence of many difficulties that face the researchers, such as the existence of connections between successive letters, the possibility that the characters are slanted, the variability of the handwritten styles and shapes, and the size of the vocabulary (lexicon size). Therefore, the research works in this area, still open, are trying to build accurate systems within satisfactory recognition rates.

Notice that an off-line approach is more complex than an on-line one, according to the presence of noise in the handwritten images and the loss of the temporal information in the writing process \cite{1}.

Handwritten recognition systems can be distinguished depending on their characteristics. A possible classification is between holistic and analytic systems. In case of a small and limited lexicon size, up to 100 classes (words), a holistic approach is considered where each word has a specific model \cite{1}\cite{11}. Whereas, in case of large lexicon size, more than 100 classes, analytical approaches are considered to be more appropriate. In such approaches, a specific model is used for each character or pseudo-character (grapheme) and the word models are built by concatenating the appropriate character models \cite{6} \cite{11}. Each of these two Analytical and holistic approaches has its own advantages. Whereas the holistic approaches do not require any kind of segmentation, analytical approaches have advantages over the second one, especially in case of a new application where the analytical approaches don't require the training of the set of words of the new lexicon. Moreover, the training of the character models doesn't depend on the size of the lexicon.

![](Figure 1. Block diagram of the handwritten recognition system.){:style="width:500px;"}

The “Hidden Markov Models” (HMM) have been successfully applied in the speech recognition field \cite{2}. More recently they have been used in the handwritten texts recognition \cite{1} \cite{9} \cite{10} \cite{11} either uniquely (independently) or by coupling with other classification models e.g. artificial neural networks \cite{1} \cite{10}. The HMM structure is well appropriate to describe a word image as a sequence of observations. While some approaches need pre-segmentation, the use of HMM offers implicit segmentation in the recognition phase. A joint optimal segmentation and characters recognition is performed.

In the work described hereafter, an off-line handwritten recognition system based on analytical approach has been built. Language independent features are extracted from the words' images. These features are then passed to an HMM based classifier that identifies the written words based on the maximum likelihood criterion. This system has been experimented on the benchmark database IFN/ENIT for Tunisian villages’ names \cite{5}. An overview of the proposed handwriting recognition system is
shown in Figure 1. Besides, presenting the baseline system, this paper stresses the importance of the choice of the features, especially the introduction of the baseline, in the performance of the system.

In section 2 we present the preprocessing and the base lines detection phase. The section 3 describes the fragmentation and the feature extraction step to provide the sequence of observations to the HMM. In section 4 we provide a brief description of the HMM-based classification Module, and we present the obtained experimental results in section 5. Finally, section 6 draws some conclusions and perspectives for our future works.

2. PREPROCESSING AND BASELINE DETECTION

In the building of an automatic off-line handwritten recognition system, many phases need to be implemented in order to achieve high accuracy. One of the most important required steps is that of preprocessing and normalization. The preprocessing phase may consist of many sub-phases such as image binarization, noise removal, slant estimation and normalization, baselines detection etc. The principle preprocessing steps were already fulfilled during the IFN/ENIT benchmark database development. In this section we will bring about the baselines estimation approach.

In the most of handwritten cases, some strokes in a word may extend above or below the middle zone of the word. Such letter components are named ascenders and descenders respectively. Examples of letters that contain such strokes are: ‘j’, ‘g’, ‘f’, ‘t’ in the Latin case, and “ج” “ه” “ث” “خ” in the Arabic case. Hence, the middle zone of a word that doesn’t contain ascenders and descenders is bounded by upper and lower baselines. The Figure 2 provides an example of baselines.

The detection of the baselines, especially the lower baseline, is essential for the subsequent steps like skew correction, slant normalization and feature extraction. Therefore, in our preprocessing phase we integrate a routine to estimate the upper and the lower baselines of a handwritten word. Our approach is based on the algorithm described in [7] with few alterations. Actually we are not interested in both baselines. The algorithm is based on the horizontal projection histograms that are computed with respect to the horizontal pixels density. In our case we consider that the peak line, or the line that contains the maximum density of foreground pixels, to be the line of writing. This line of writing is named in our works the lower baseline. This is justified with the Arabic written language where most of the letters have a lot of pixels on the lower baseline.

3. FEATURE EXTRACTION

The feature extraction is one of the most important parts of any classification system, and it plays an important role in improving the recognition rate. The global performance of the recognition system depends largely on the features used.

Figure 3: Sliding window divides the word into vertical frames.

The feature extraction process in our approach is based on the vertical sliding window techniques. The vertical sliding window shifts across the word image from right to left (left to right for Latin scripts). At each step, the system isolates a frame (a vertical strip) from which we compute a features vector. The width of the resulted frame is a system parameter and the height is determined automatically according to the whole word image height. The result is a sequence of overlapping frames. The overlap from one frame to the next is currently a system parameter also (from 0 pixels to frame-width-1 pixels). Moreover, each frame is horizontally divided into equal cells, the number of cell varies according to the word image height.

The computed features are based on the density and the distribution of the foreground pixels (black pixels) in the word image. We distinguish between two kinds of features following their dependence on the baselines. In our system we used the following features:

- The density of foreground pixels in the vertical columns in each frame.
- The densities of foreground pixels in the cells of the vertical column in each frame.
- The number of passing between two consecutive cells of different density of foreground pixels.
- The normalized difference between the actual frame center of gravity and the center of gravity of the precedent frame
- The density of foreground pixels over and under the lower baseline.
- The vertical position of the center of gravity of the black pixels with respect to the upper and the lower baselines.

The result is a 16-features vector per frame. First and second derivations may be associated to the features vectors, leading to parameters vectors of 32 and 48 dimensions.
respectively. Note that the features are not specific to a particular type of handwritten script (loops, dotes, curves,...) that require particular recognition.

4. HMM-BASED CLASSIFIER

The use of Hidden Markov Models HMMs in developing handwritten recognition systems has several advantages, such as automatic training on non-segmented words, and the simultaneous segmentation – recognition [10]. Therefore, in case of HMMs modeling, there is no need to apply any character-level pre-segmentation process. In fact, a pre-segmentation-free approach is quite important, especially in case of cursive handwritten, e.g. Arabic handwritten texts, where the characters are often connected.

The recognition (classification) module in our recognition system is based on the HMM modeling of each character in the lexicon. We use the HCM toolkit [2] in our system.

4.1. Classifier components

The HMM-based classifier contains two modules:
- The HMM modeling and training module,
- The recognition module.

The characters-modeling and training module uses the features vectors and the information coming from the grammatical and the lexical description to estimate the characters-HMM models. A segmental Expectation Maximization (EM) algorithm, also known as Viterbi algorithm, is used in this module.

In the recognition (classifier) module, the features vectors extracted from an image are passed to a network of lexicon entries formed of the character HMM models. The characters sequence providing the maximum likelihood by using Viterbi algorithm identifies the recognized entry [4].

4.2. Character-HMM structure

Our system is characters-HMM based. Each character is modeled by an HMM model. Each HMM model is characterized by the number of states, the transitions between states, and the probability density functions associated to the different states. The determination of this structure is an identification problem. Based on a number of experiments, we specify 4 states for each character-model, and for each state we specify 3 transitions: a self-transition, a transition to the next state, and a transition that permits the skipping of a single state. The probability distribution in each state is a mixture of 3 Gaussian distributions. Therefore, the model for each word in the lexicon is built by concatenating the appropriate character models.

5. EXPERIENCES AND RESULTS

5.1. IFN/ENIT database

To evaluate the performance of our recognition system, we use in the experiments a new benchmark database IFN/ENIT. The IFN/ENIT contains a total of 26459 handwritten words of 946 Tunisian town/villages names (a city name can be single or compound words) written by different writers [5].

In order to model all the Arabic characters, we built up to 159 character models. In fact, an Arabic character may have different shapes according to its position in the word. Other models are specified for spaces as well as for characters with additional marks. Therefore, a word model is built up by concatenating the appropriate character models.

Due to the fact that the occurrence of examples in the IFN/ENIT database is different from one word to another, our lexicon consists of 200 villages/town names related to a dataset of 19654 entries. This dataset is divided into 4 subsets known as subsets a, b, c, d.

5.2. Baseline results

In our experiments, we try to emphasize the importance of the features related to the estimated positions of the baselines in the preprocessing phase. Therefore, we have obtained different recognition rates according to the type of the features extracted from the word images. For every type of feature 4 experiments are conducted by taking alternatively one subset for testing and the remaining three subsets for training. This leads to a nonparametric estimation of the error rate.

The Table 1 shows the experimental results of the performance evaluation of our recognition system using all the 16 features in the feature vector. Preliminary experiments shows that only the first-order derivatives must be considered. This leads to a recognition (resp. error) rate of 87.02% (12.98%).

<table>
<thead>
<tr>
<th>TEST data set</th>
<th>Training data set</th>
<th>Test data set</th>
<th>Rec. rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>data size</td>
<td></td>
<td>size</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>bcd</td>
<td>a</td>
<td>85.42 %</td>
</tr>
<tr>
<td>2</td>
<td>acd</td>
<td>b</td>
<td>87.9 %</td>
</tr>
<tr>
<td>3</td>
<td>abd</td>
<td>c</td>
<td>86.84 %</td>
</tr>
<tr>
<td>4</td>
<td>abc</td>
<td>d</td>
<td>87.93 %</td>
</tr>
</tbody>
</table>

In Table 2 we provides the results after removing some features relative to the detected baseline. The recognition (error) rate drops (increases) to 83% (17%). This shows that the features based on the baselines detection information bring a significant improvement (error rate reduced by 23.5%) to the recognition performance.

A preliminary error analysis shows different causes for wrong classifications according to the nature of the Arabic cursive handwritten words.

In fact, the diactrical marks such as the dots (e.g. in case of the letters: \( \dddot{\text{.}} \), \( \dddot{\text{.}} \), \( \dddot{\text{.}} \) ..., the ligatures in the descenders of some letters (e.g. in case of the letters: \( \dddot{\text{.}} \), \( \dddot{\text{.}} \), \( \dddot{\text{.}} \) ...), often are not at
the exact position on top or under the main part of the letter. Furthermore, the descenders of a letter can be extended under one or more letters of the same word (Figure 4). Therefore, the vertical slicing approach will eventually split letters from their diacritical marks, and thus reduce the character recognition accuracy.

Table 2. Recognition results without baseline estimation.

<table>
<thead>
<tr>
<th>TEST data set</th>
<th>Training data set</th>
<th>Test data set</th>
<th>Rec. rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>size</td>
<td>size</td>
<td>size</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>bcd 8672</td>
<td>a 2854</td>
<td>82.31 %</td>
</tr>
<tr>
<td>2</td>
<td>acd 8628</td>
<td>b 2898</td>
<td>83.92 %</td>
</tr>
<tr>
<td>3</td>
<td>abd 8738</td>
<td>c 2788</td>
<td>82.12 %</td>
</tr>
<tr>
<td>4</td>
<td>abc 8540</td>
<td>d 2986</td>
<td>83.64 %</td>
</tr>
</tbody>
</table>

Figure 4. Descenders and dots.

6. CONCLUSIONS AND PERSPECTIVES

This work presented a system for offline cursive handwritten recognition. The system is based on character-HMMs approach. In the pretreatment phases the baselines positions are detected, and the extracted features are script (language)-independent.

Results on the new benchmark database IFN/ENIT are presented, and show the improvement of the recognition rate coming from the using of the proposed features that are computed according to the detected positions of the baselines. In [6], the features are extracted by directly applying a KL transformation on the pixels. Although we cannot compare to the results in [6] (the lexicon in our work is smaller), however, the performance we obtain using our features seems to be promising. The accuracy of the baselines detection algorithm is quite satisfactory, and the recognition rates obtained so fare are promising.

Many points are yet to be achieved such as proposition of new discriminant and robust features, such as those resulting from “Vectorization Quantization” (VQ) process and “Linear Discriminate analysis” (LDA). A major perspective consists in taking into account the skew of the baseline in the feature extraction.

Acknowledgment

This work is partially supported by the EC project NEMLAR. The authors are grateful to Drs. M. Pechwitz, S. Maddouri, V. Maegner and N. Ellouze for providing the IFN/ENIT database. Thanks go also to Dr. N. Nassif, Dr. L. Likforman and Ms. R. Bayeh, for their precious help.

7. REFERENCES