Recognition of Printed Arabic Words with Fuzzy ARTMAP Neural Network

Adnan Amin[1] and Nabeel Murshed[2]
[1]School of Computer Science and Engineering
University of New South Wales, Sydney-Australia
amin@cse.unsw.edu.au
[2] Center for Information Science and Processing
Tuiuti University of Paraná, Curitiba, Brazil
nmurshed@cognus.eti.br, nmurshed@utp.br

Abstract

This paper presents a new method for the recognition of Arabic text using global features and Fuzzy ARTMAP neural network. The method is divided into three major steps. The first step is digitization and pre-processing to create connected component. The second step is concerned with feature extraction, where global features of the input word are used to extract features such as number of subwords, number of peaks within the subword, number and position of the complementary character, etc., to avoid the difficulty of segmentation stage. The third step is the classification and is composed of a single Fuzzy ARTMAP. The method was evaluated with 3255 images of 217 Arabic words with different fonts (each word has 15 samples), and the mean correct classification rate was 95.25%.

1. Introduction

For the past three decades, there has been increasing interest among researchers in problems related to the machine simulation of the human reading process. Intensive research has been carried out in this area with a large number of technical papers and reports in the literature devoted to character recognition. This subject has attracted immense research interest not only because of the very challenging nature of the problem, but also because it provides the means for automatic processing of large volumes of data in postal code reading [1,2], office automation [3,4], and other business and scientific applications.

Different approaches covered under the general term *character recognition* fall into either *on-line* or *off-line* category, each having its own hardware and recognition methodology.

In on-line character recognition systems, the computer recognizes symbols as they are drawn. The most common writing surface is the digitizing tablet, which operates through a special pen in contact with the surface of the tablet and emits the coordinates of the plotted points at a constant frequency. Breaking contact prompts the transmission of a special character. Thus, recording on the tablet produces strings of coordinates separated by signs indicating when the pen has ceased to touch the tablet surface.

Off-line recognition is performed after the writing or printing process is completed. Optical Character Recognition, *OCR*, deals with the recognition of optically processed characters rather than magnetically processed ones. In a typical OCR system, input characters are read and digitised by an optical scanner. Each character is then located and segmented and the resulting image is fed into a preprocessor for smoothing, noise reduction, and size normalisation. Off-line recognition can be considered the most general case in that no special device is required for writing and signal interpretation is independent of signal generation, as in human recognition.

Many works have been concerned with the recognition of Latin, Chinese and Japanese characters. However, although almost a third of a billion people worldwide, in several different languages, use Arabic characters for writing, little research progress, in both on-line and off-line has been achieved towards the automatic recognition of Arabic characters. This is a result of the lack of adequate support in term of funding, and other utilities such as Arabic text database, lexicon, etc. and of the cursive nature of its writing rules.

There are two strategies which have been applied to printed and handwritten Arabic character recognition. These can be categorized as *holistic* and *analytic*. In holistic strategies the recognition is performed on the

0-7803-5529-6/99/$10.00 ©1999 IEEE

whole representation of the word and there is no attempt to identify its characters individually. These strategies were originally introduced for speech recognition and can fall into two categories:

- Methods based on distance measurements using Dynamic Programming [5].
- Methods based on a probabilistic framework (Hidden Markov Models) [6,7].

In analytical strategies words are not considered as a whole, but as sequences of small size units and the recognition is not directly performed at the word level but at an intermediate level dealing with these units, which can be graphemes, segments, pseudo-letters, etc. [8,9]. Surveys on Arabic recognition can be found in [10,11].

This work adopts the holistic strategy and proposes the use of global features and the Fuzzy ARTMAP neural network for recognizing printed Arabic words. Section 2 presents the characteristics of Arabic writing. In sections 3 and 4, the proposed method is presented. Experimental results are described in section 5.

2. General Characteristics of the Arabic Writing

Arabic writing is similar to English in that it uses letters (which consist of 29 basic letters), numerals, punctuation marks, as well as spaces and special symbols. It differs from English, however, in its representation of vowels since Arabic utilizes various diacritical markings. The presence and absence of vowel diacritics indicates different meanings in what would otherwise be the same word. If the word is isolated, diacritics are essential to distinguish between the two possible meanings. If it occurs in a sentence, contextual information inherent in the sentence can be used to infer the appropriate meaning. In this paper, the issue of vowel diacritics is not treated, since it is more common for Arabic writing not to employ these diacritics. Diacritics are only found in old manuscripts or in very specific areas.

The Arabic alphabet is represented numerically by a standard communication interchange code approved by the Arab Standard and Metrology Organization (ASMO). Similar to the American Standard Code for Information Interchange (ASCII), each character in the ASMO code is represented by one byte. An English letter has two possible shapes, upper and lower cases. The ASCII code provides separate representations for both of these shapes, whereas an Arabic letter has only one representation in the ASMO table. This is not to say, however, that the Arabic letter has only one shape. On the contrary, an Arabic letter might have up to four different shapes, depending on its relative position in the text. For instance, the letter (A'in) has four different shapes: at the beginning of the word (preceded by a space), in the middle of the word (no space around it), at the end of the word (followed by a space), and in isolation (preceded by an unconnected letter and followed by a space). These four possibilities are exemplified in Figure 1.

In addition, different Arabic characters may have exactly the same shape, and are distinguished from each other only by the addition of a *complementary character*[1] . These are normally a dot, a group of dots or a zigzag-shape character known as *hamza*. This may appear on, above, or below the baseline, and can be positioned differently, for instance, above, below or within the confines of the character. Figure 2 depicts two sets of characters, the first set having five characters and the other set three characters. Clearly, each set contains characters which differ only by the position and/or the number of dots associated with it. It is worth noting that any erosion or deletion of these complementary characters may result in a misrepresentation of the character.

Arabic writing is cursive and is such that words are separated by spaces. However, a word can be divided into smaller units called *subwords*[2]. Some Arabic characters are not connectable with the succeeding character. Therefore, if one of these characters exists in a word, it divides that word into two subwords. These characters appear only at the tail of a subword, and the succeeding character forms the head of the next subword. Figure 3 shows three Arabic words with one, two, and five subwords. The first word consists of one subword which has nine letters; the second has two subwords with three and one letter, respectively. The last word contains five subwords, each consisting of only one letter.

In general, Arabic writing can be classified into typewritten (Naskh), handwritten (Ruq'a) and artistic (or decorative Calligraphy, Kufi, Diwani, Royal, and Thuluth) styles as shown in Figure 4. Handwritten and decorative styles usually include vertical combinations of short strokes called *ligatures*. This feature makes it difficult to determine the boundaries of the characters. Furthermore, characters of the same font have different sizes (i.e. characters may have different widths even though the two characters have the same font and point size). Hence, word segmentation based on a fixed width cannot be applied to Arabic.

3. Digitization and Preprocessing

The first phase in our character recognition system is digitization. Document to be processed are first scanned and digitized. The algorithm adopted in this paper is

[1] A portion of a character that is needed to complement an Arabic character.

[2] A portion of a word including one or more connected characters.

similar to that which appears in [12]. A 300 dpi scanner is used to digitize the image in this phase and the output is a standard binary formatted image (bmp format).

After the digitization is completed, the connected components must be determined. Connected components are rectangular boxes bounding together regions of connected black pixels. The objective of the connected component stage is to form rectangles around distinct components on the page, whether they arer characters or images. These bounding rectangles then form the skeleton for all future analysis on the page.

The algorithm used to obtain the connected components is a simple iterative procedure which compares successive scanlines of an image to determine whether black pixels in any pair of scanlines are connected together. Bounding rectangles are extended to enclose any groupings of connected black pixels between successive scanlines. Figure 4 demonstrates this procedure.

Each scanline in Figure 4 is 14 pixels in width (note that a pixel is represented by a small rectangular block), the bounding rectangles in Figure 4.a just enclose the black pixel of that scanline, but for each successive scanline the bounding boxes increase to include the black pixels connected to the previous scanline. Figure 4.c also points out that a bounding box stops growing in size only when there are no more black pixels on the current scanline joined onto black pixels of the previous scanline.

4. Global Word Feature

In this work, we have used the second approach to extract the proper characteristic of Arabic characters. Seven types of global features have been extracted such as: number of subwords, number of peaks of each subwords, number of loops of each peak, number and position of complementary characters, the height and width of each peak. Feature extraction algorithm can be summarized into the following steps:

Step 1: Loop detection: Loops are detected simply as being the inner contours (Figure 5) obtained by running the contour tracing algorithm. The tracing algorithm traces outer contour of an object and then it traces the inner contours of the object.

Step 2: Determine the number of peaks within the sub-word: In all printed Arabic characters, the width at a connection point is much less than the width of the beginning character. Therefore, the baseline is a medium line in the Arabic word in which all the connections between the successive characters take place. If a vertical projection of bi-level pixels is performed on the word according to the following equation:

$$v(j) = \sum_{i} w(i,j) \qquad (1)$$

where $w(i, j)$ is either zero or one and i, j index the rows and columns, respectively, then the peak point will have a sum greater than the average value AV given by:

$$AV = (1 / Nc) \sum_{j=1}^{Nc} Xj \qquad (2)$$

where Nc is the number of columns and X_j is the number of black pixels of the j^{th} column.

Step 3: Smooth the histogram by using the averaging scheme where each point in the histogram is replaced by the average of itself and the two points on either side of it.

$$X_i = (X_{i-1} + X_i + X_{i-2}) / 3 \qquad (3)$$

Step 4: Complementary characters: This feature plays very important role to distinguish between characters having the same shape. The complementary characters could be either a zigzag or a a group of dots (1, 2 or 3). These can be above or below the baseline. Similar characters or subwords with dots have different meaning and pronunciation if the a group of dots below or above the base line.

Step5: Compute the height and width of each peak whether large of small.

5. Experimental Results

The classification stage consists of one Fuzzy ARTMAP neural network. The database contains 3255 images of 217 words (15 samples/word). Each image was digitized and binarized with 300 dpi scanner. For each word, the samples were divided into training, S^{tr}, and evaluation sets, S^{ev}, each of which contained 10 and 5 images, respectively. Based on the above, the training and evaluation sets, contained 2170 and 1085 samples, respectively.

5.1 Estimation of Classification Rate

The classification rate was measured according to the following experimental procedures:

start
  for each trial $k = 1, 2, ... 30$;
    for each word W_i, $i = 1, 2, ..., 217$;
      select randomly 10 samples that have not been selected before;
      form the training S^{tr};
      form the evaluation set S^{ev} from the remaining 5 samples;
    end for;

train the network with S^t and evaluate it with S^{cv};
record and accumulate the classification errors;
end for;
calculate the mean error;
end;

The parameters for the Fuzzy ARTMAP were: ρ = 0.92, β = 0.5, and α = 0.01. The epoch size and training iteration were, respectively, 500 and 100000. The input vectors to the Fuzzy ARTMAP are binary vectors and consisted of 124 elements and 9 elements, respectively. The 124-element vector is the feature vector, whereas the 9-element vector is the class label (0, 1, ..., 216). It should be noted that the network parameters were dertermined using a small subset of the database.

The classification rate were calculated according to the Correct Classification Rate (*CCR*), which indicates the percentage of words classified correctly by the Fuzzy ARTMAP. With the above experimental procedures, the the mean *CCR* for the Fuzzy ARTMAP was 95.25%.

6. Discussion and Conclusion

In this paper we have presented a new method for recognizing printed Arabic words. It is based on the holistic approach in which the recognition is performed at the word level, which is inexpensive for feature extraction. As mentined above, each word is presented by seven global features. These features formed the 124-element input vector to the Fuzzy ARTMAP.

From the obtained results, we may conclude that the proposed method is very effective; and that the Fuzzy ARTMAP is an appropriate choice for the proposed method. However, we are very well aware that further studies are required to increase the classification rate. Our future work will focus on obtaining a large database, and optimizing the feature representation.

7. References

[1] Harmon L. D, Automatic recognition of printed and script, Proc. IEEE, 60, 10, 1165–1177, (1972).

[2] Spanjersberg A. A, Experiments with automatic input of handwritten numerical data into a large administration system, IEEE Trans. Man Cybern. 8, 4, 286–288, 1978.

[3] Focht L. R and Burger A, A numeric script recognition processor for postal zip code application, Int. Conf. Cybernetics and Society, 486–492, 1976.

[4] Schuermann J, Reading Machines, 6th Int. Conf. on Pattern Recognition, 741–745, 1982.

[5] M. Khemakhem and M. C. Fehri, Recognition of Printed Arabic charcters by comparaison dynamique, Proc. First Kuwait Computer Conference, pp. 448–462, 1989.

[6] R. Schwartz, C. LaPre, J. Makhoul, C. Raphael, and Y. Zhao. Language independent ocr using a continuous speech recognition system. 13th International Conference on Pattern Recognition, vol. C, pages 99-103, Vienna, Austria,1996.

[7] N. BenAmara and A. Belaid. Printed PAW Recognition Based on Planar Hidden Markov Models. In 13th International Conference on Pattern Recognition, Vol. B, Vienna, Austria, 1996.

[8] A. Amin and J. F. Mari, Machine recognition and correction of printed Arabic text, *IEEE Trans. Man Cybern.* **9**(1), 1300–1306, 1989.

[9] H. Almuallim and S. Yamaguchi, A method of recognition of Arabic cursive handwriting, *IEEE, Trans. Pattern Anal. and Machine Intell.* **PAMI-9**, 715–722, 1987.

[10] B. Al-Badr and S. Mahmoud, Survey and bibliography of Arabic optical text recognition, Signal Processing 41 , 49-77,1995.

[11] A. Amin, Off-line Arabic character recognition: The State of the art, Pattern Recognition **31 (5)**, pp. 517-530, 1998

[12] A. Amin and H. B. Al-Sadoun, A new structural technique for recognizing printed Arabic text, *Int. J. of Pattern Recognition and Artif. Intell.* **9**, 1 (1995) 101–125.

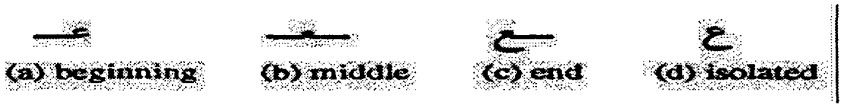

Figure 1. Different shapes of the Arabic letter " A'in "

Figure 2. Arabic characters differing only with regard to the position and number of associated dots.

Figure 3. Arabic words with constituent subwords.

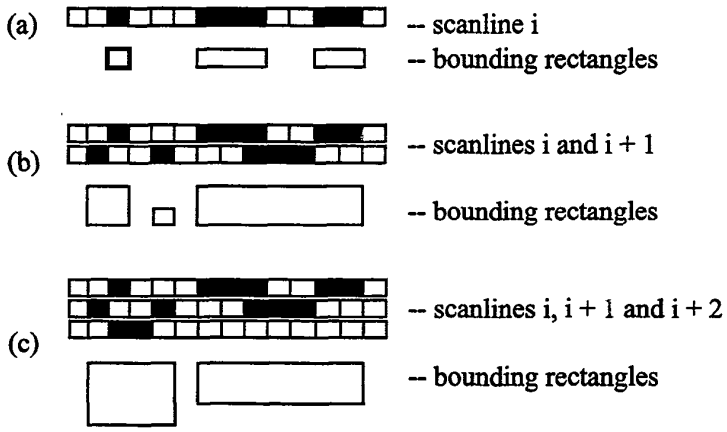

Figure 4. The process of building connected components from image scanlines.

Figure 5. The inner and outer contour of an Arabic word.