July 2004, Vol.19, No.4, pp.538–543 J. Comput. Sci. & Technol.

Printed Arabic Character Recognition Using HMM

Abbas H. Hassin, Xiang-Long Tang, Jia-Feng Liu, and Wei Zhao

Computer Science Department, Harbin Institute of Technology, Harbin 150001, P.R. China

E-mail: Abbashh2002@yahoo.com

Received July 10, 2003; revised November 21, 2003.

Abstract The Arabic Language has a very rich vocabulary. More than 200 million people speak this language as their native speaking, and over 1 billion people use it in several religion-related activities. In this paper a new technique is presented for recognizing printed Arabic characters. After a word is segmented, each character/word is entirely transformed into a feature vector. The features of printed Arabic characters include strokes and bays in various directions, endpoints, intersection points, loops, dots and zigzags. The word skeleton is decomposed into a number of links in orthographic order, and then it is transferred into a sequence of symbols using vector quantization. Single hidden Markov model has been used for recognizing the printed Arabic characters. Experimental results show that the high recognition rate depends on the number of states in each sample.

Keywords pattern recognition, off-line Arabic character recognition, feature extraction, hidden Markov models

1 Introduction

The Arabic language has a very rich vocabulary. More than 200 million people speak this language as their native language, and over 1 billion people use it in several religion-related activities. Arabic character recognition, both printed and handwritten, is one of the most challenging tasks and exciting areas of research in Optical Character Recognition (OCR). Indeed, despite the growing interests in this field, no satisfactory solution is available. The main reason is the special and complex characteristics of Arabic text. For this reason, Arabic character recognition remained as an untouched field until 1980[1]. Subsequently, a number of Arabic character recognition systems were proposed[2–4].

Amin[5] segments a printed text into words and a word into characters. Then, each character is divided into as many as seven primitives using the length of the rows (columns) of the horizontal and vertical projections as features. The complementary characters are identified by using their density. The projection histogram is coded as a string whose symbols are indications to show where a row (column) is of length zero, greater than the average for that projection, or less than the average.

Khella[6] uses a tree structure to group the Arabic character sets based on the number and location of dots and holes. Then he uses a statistical classifier to identify the characters of the same group.

Some systems do not follow the classical paradigm of pattern recognition which is feature extraction followed by classification. Al-Badr and Haralick use a set of 30 shape primitives as features. They detect those shape primitives on a page image using mathematical morphology operations. A character is defined as a set of primitives at a certain configuration relative to one another. Whenever the primitives of a character are found with the right configuration, the character is detected[7].

The main characteristics of Arabic text can be summarized as follows.

• Arabic text (printed or handwritten) is cursive and written from right to left. Arabic letters are normally connected to each other on the baseline.

• Arabic text uses letters (which consist of 28 basic forms), ten Hindi numerals, punctuation marks, as well as spaces and special symbols. Table 1 shows the Arabic letters.

• Some of the Arabic letters are located under the baseline (for example, ر (raa), ز (zay), و (waw)).

• Arabic letters might have up to four different forms (beginning, middle, end, and isolated) depending on their relative position in the word (for example, غـ, ـغـ, ـغ, ـغ respectively). This feature increases the number of classes from 28 to 100. Actually, a new character is created when Alif (ا) is written immediately after the letter Lam (ل). This kind of new character increases the number of classes to 120 (see Table 2).

• Several Arabic letters have exactly the same primary part. However, they are distinguished from each other by the addition of dots in different

*Correspondence

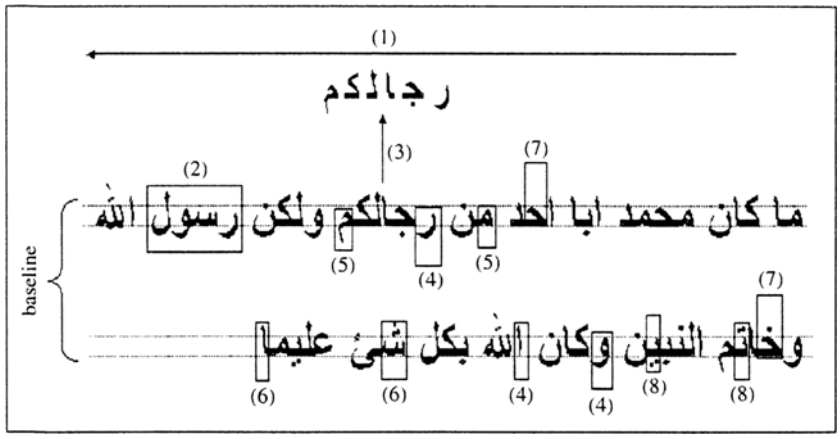

Fig.1. Characteristics of Arabic text. (1) Written from right to left. (2) One Arabic word includes three cursive subwords. (3) A word consists of six characters. (4) Some characters are not connectable on the left side with the succeeding character. (5) The same character with different shapes depends on its position in the word. (6) Different characters with different sizes. (7) Different characters with a different number of dots. (8) Different characters have the same number of dots but different positions of dots.

locations.

• Arabic characters do not have a fixed size. The width and height of a character vary according to its position in the word.

• Arabic writing is cursive and words are separated by spaces. Some Arabic letters are not connectable with the succeeding letter. Therefore, if one of these letters exists in a word, it divides that word into two sub-words. These letters appear only at the tail of a sub-word and the succeeding letter forms the head of the next sub-word.

Fig.1 introduces a brief summary of the main characteristics of Arabic characters. This figure includes 8 points, each point describes an individual feature.

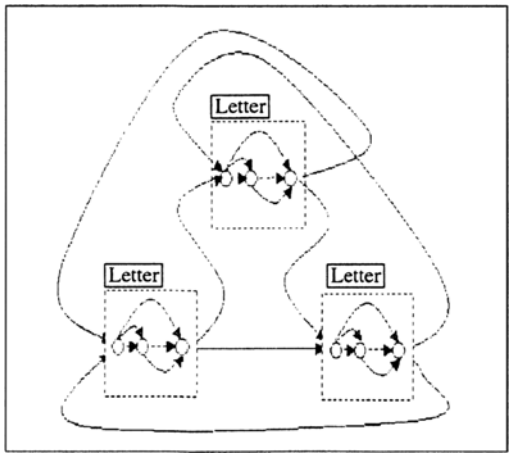

Fig.2. Single HMM topology.

2 Hidden Markov Models

A character recognition system is composed of a collection of algorithms drawn from a wide variety of disciplines such as signal processing and statistical pattern recognition. Different recognizers rely to varying degrees on the signal processing front end to convert the character/word image to some form of parametric representation for further analysis and processing. In this section, we present the technique of Hidden Markov Models.

Hidden Markov Model (HMM) is a powerful tool in the field of signal processing[8,9]. HMM has been successfully used in speech recognition[10]. Recently, the application of HMM has been extended to include word recognition[11,12]. This is due to the similarities between speech and written words since they both involve co-articulation, which suggests the processing of symbols with ambiguous boundaries and variations in appearance. Fig.2 depicts a single HMM topology.

2.1 Mathematical Definition of HMM

The purpose of this section is to describe the HMM definition in detail. The definition of an HMM must specify the model topology, the transition parameters and the output distribution parameters. An HMM is a stochastic process with an underlying finite-state structure. Each of these states is associated with a random function. Within a state the signal possesses some measurable, distinctive properties. Within a discrete pe-

riod of time, the process is assumed to be in some state and an observation is generated by the random function of that state. The underlying Markov chain changes to another state with the transition probability of the current state. The sequence of states is hidden; only the sequence of observations produced by the random function of each state can be seen.

Table 1. Arabic Letters

No.	Letters	IF	BF	MF	EF
1	Alif	ا	ا	ـا	ـا
2	Baa	ب	بـ	ـبـ	ـب
3	Taa	ت	تـ	ـتـ	ـت
4	Thaa	ث	ثـ	ـثـ	ـث
5	Jeem	ج	جـ	ـجـ	ـج
6	Hha	ح	حـ	ـحـ	ـح
7	Kha	خ	خـ	ـخـ	ـخ
8	Dal	د	د	ـد	ـد
9	Thal	ذ	ذ	ـذ	ـذ
10	Raa	ر	ر	ـر	ـر
11	Zay	ز	ز	ـز	ـز
12	Seen	س	سـ	ـسـ	ـس
13	Sheen	ش	شـ	ـشـ	ـش
14	Sad	ص	صـ	ـصـ	ـص
15	Dhad	ض	ضـ	ـضـ	ـض
16	Tta	ط	طـ	ـطـ	ـط
17	Ttha	ظ	ظـ	ـظـ	ـظ
18	Ain	ع	عـ	ـعـ	ـع
19	Ghain	غ	غـ	ـغـ	ـغ
20	Faa	ف	فـ	ـفـ	ـف
21	Gaf	ق	قـ	ـقـ	ـق
22	Kaf	ك	كـ	ـكـ	ـك
23	Lam	ل	لـ	ـلـ	ـل
24	Meem	م	مـ	ـمـ	ـم
25	Noon	ن	نـ	ـنـ	ـن
26	Ha	ه	هـ	ـهـ	ـه
27	Waw	و	و	ـو	ـو
28	Yaa	ي	يـ	ـيـ	ـي

IF: Isolated Form, BF: Beginning Form
MF: Middle Form, EF: End Form

Basically, the system transits from one state to another depending on a set of probabilities associated with each state. In general, a system transits from state q_i at time t to a state q_j at time $t+1$, $t = 1, 2, \ldots$ and $i, j = 1, 2, \ldots, N$.

The most important and difficult element to be decided is the number of states, N, in the model. As mentioned above, at each time the model makes a change from one state to another or may remain in the same state. The transition is based on a transition probability associated with the previous state.

The number of observation symbols is another important element in the model. The observation symbols, M, correspond to the physical output of the system being modeled. For some applications like speech recognition and character recognition, the observations are continuous and are produced as vectors. In this case, the vectors are quantized into one of the allowable sets using Vector Quantization (VQ)[13].

Formally, the rest of the elements of an HMM are defined as follows.

1. *The initial state probability.* This is the probability of being in state q_i at time t.

$$\pi = \{\pi_i = P(q_i \text{ at } t = 1)\}$$

2. *The state transition probability.* This is the probability of being in state q_i at time t, then transiting to state q_j at time $t+1$.

$$A = \{a_{ij} = P(q_j \text{ at } t+1 | q_i \text{ at } t)\}$$

3. *The observation symbol probability.* This is the probability of observing symbol v_k while the model is in state q_i at time t.

$$B = \{b_i(k) = P(v_k \text{ at } t | q_i \text{ at } t)\}$$

Generally, a complete model of an HMM requires specifying many parameters. These parameters are N and M as mentioned before, the length of the observation sequence (T), an observation sequence denoted as $O = (o_1 o_2 \ldots o_T)$, the observation symbol denoted as $V = (v_1 v_2 \ldots v_M)$, and three sets of probability measures π, A, B. The compact notation used to refer to an HMM is $\lambda(\pi, A, B)$.

Table 2. Additional Characters from Complementary (Hamza and Madda)

No	Letters	IF	BF	MF	EF
1	Alif	أ			ـأ
2	Alif	آ			ـآ
3	Alif	إ			ـإ
4	LamAlif	لا			ـلا
5	LamAlif	لأ			ـلأ
6	LamAlif	لإ			ـلإ
7	LamAlif	لآ			ـلآ
8	Waw	ؤ			ـؤ
9	Ya	ئ	ـئ	ـئـ	ـئ

Application of HMM in real-world encounters three key problems. These problems are the following:

1. Given the observation sequence $O = (o_1 o_2 \ldots o_T)$ and the model $\lambda(\pi, A, B)$, how we can compute $P(O|\lambda)$, the probability of the observation sequence.

2. Given the observation sequence $O = (o_1 o_2 \cdots o_T)$ and the model $\lambda(\pi, A, B)$, how we can choose a corresponding state sequence $Q = (q_1 q_2 \ldots q_T)$.

3. Given an HMM, how we adjust the model parameters $\lambda(\pi, A, B)$ to maximize $P(O|\lambda)$.

More details and mathematical solutions of each one of these three problems can be found in [9].

3 Implementation of HMM

The purpose of this paper is to construct a single HMM for off-line recognition of Arabic characters. The features used in the HMM are based on the arcs of the skeleton of the words to be recognized. A number of reasons motivate the proposed technique. First, we wish to segment words into characters or other primitives. Second, extracting edges from the skeleton is more reliable than actual connection points in the word. Ultimately, the extracted features are shape descriptors of the skeleton graph, so they provide a compromise between a powerful recognition and efficient extraction.

Structural features of off-line printed Arabic characters are extracted from the skeleton graph of a given word image. These features are represented as a feature vector. A feature vector needs to be encoded into one of the discrete symbols in order to reduce the computation required in the HMM-based recognition system. Vector quantization is used for this purpose. Finally, the results of vector quantization are used to form the observation sequences.

Fig.3 shows an illustration of transferring a word image into a sequence of feature vectors.

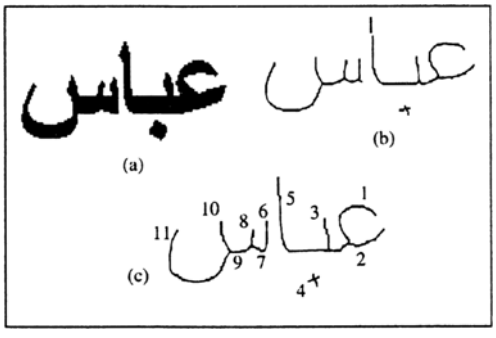

Fig.3. Illustration of transferring a word image into a sequence of feature vectors. (a) An original word image. (b) Skeleton graph of (a) after applying thinning algorithm. (c) Producing 11 feature vectors from (b).

The lexicon builds only one model for all the words and uses different paths (state sequences) through the model to distinguish one class from the others. The classification mechanism here is selecting the maximum path probability of the class over all possible paths. This is called a path discriminating HMM in which states are clear.

The HMM is formed from elementary units. These units include the 28 basic letters mentioned in Table 1, four additional letters (i.e., ة, ى, ء, لا) and 19 two-letter combinations. These combinations consist of two letters that are not separated by any feature point thus it is impossible to decompose the main stroke into more than one link. The 19 combinations include جا, حا, خا, جى, حى, خى, با, تا, ثا, نا, يا, بي, تي, ثي, ني, يى, لي, كا and كى.

Each elementary unit represents at least one letter and is structured as a left-to-right HMM. The number of states in that model is relative to the number of links of the letter or the two-letter combination.

Fig.4 shows the two-letter combination of حا.

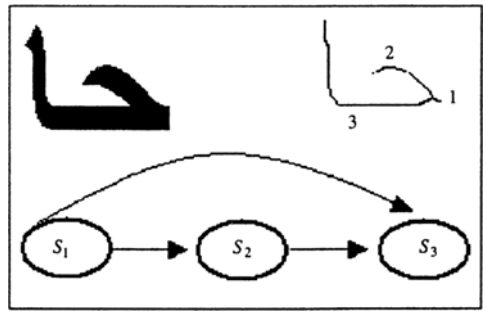

Fig.4. Two-letter combination and its elementary unit representation.

3.1 Symbol Probabilities

The VQ technique is used for assigning symbols in the HMM approach. The symbol probability computation is completed in training stage. In the recognition stage each feature vector is assigned to the nearest codebook symbol centroid based on the Euclidean distance measure. The symbol probability can be calculated as follows:

$$b_i(k) = \frac{\text{No. of times in state } i \text{and observing symbol } v_k}{\text{Total number of times in state } i}$$

3.2 State Probabilities

The initial state probability can be computed based on the location of the state within its ele-

mentary unit and the type of elementary unit, that is, whether it represents a single letter or two-letter combination.

If the state is the first within a unit that represents a single letter, then the initial probability is equal to P_0 of that letter. If the elementary unit represents two-letter combination, the initial probability of the first state equals the product of P_0 of the first letter and the letter transition probability from the first letter to the second letter $P_1(\alpha \to \beta)$. Otherwise, the initial state probability equals zero.

$$\pi_d = \begin{cases} P_0(\alpha), & i \text{ is the 1st state in the elementary unit } \alpha; \\ P_0(\alpha) \times P_1(\alpha \to \beta), & i \text{ is the 1st state in the elementary unit } \alpha\beta; \\ 0, & \text{Otherwise.} \end{cases}$$

The state transition probability within the same elementary unit is calculated as follows:

$$P_{s_i \to s_j}(\alpha) = P(q_j \text{ at } t+1 | q_i \text{ at } t)$$

The state transition probability between any two states in the model is calculated accordingly.

$$a_{i,j} = \begin{cases} 0, & i \ \& \ j \text{ are middle states in different letters;} \\ 0, & i \ \& \ j \text{ are in the same letter, } i \geqslant j; \\ P_{s_i \to s_j}(\alpha), & i \ \& \ j \text{ are in the same letter, } i < j; \\ P_1(\alpha \to \beta), & i \text{ is the last state in } \alpha, \text{ and } j \text{ is the first state in } \beta. \end{cases}$$

4 Evaluation Results

Each image first passed five stages: thinning, skeleton modification, links and loops extraction, feature extraction, and VQ. In this work, we applied the thinning algorithm after normalizing all the words. Therefore, we got high thinning rate. This resulted in a sequence of observations that were introduced to the HMM.

قيل

System output		$P(O\|\lambda)$
	قيل	8.12×10^{-11}
†	فيل	3.72×10^{-11}
	قبل	6.51×10^{-12}
‡	فبل	4.31×10^{-12}
	قيل	1.84×10^{-12}

Fig.5. System's output of the sample "قيل" from simplified Arabic font.*

علي

System output		$P(O\|\lambda)$
	علي	7.43×10^{-9}
	على	3.89×10^{-9}
	علي	9.57×10^{-10}
†	حلي	4.65×10^{-11}
†	غلي	1.54×10^{-12}

Fig.6. System's output of the sample "علي" from Thuluth font.

العرب

System output		$P(O\|\lambda)$
†	الحرب	5.13×10^{-16}
	العرب	3.94×10^{-16}
‡†	العزب	9.57×10^{-17}
	العرب	6.03×10^{-17}
‡†	العدب	5.54×10^{-17}

Fig.7. System's output of the sample "العرب" from Arabic traditional font.

حنون

System output		$P(O\|\lambda)$
	حنون	9.34×10^{-11}
‡	حنوت	1.56×10^{-11}
‡	خنون	1.07×10^{-12}
‡†	خيون	1.26×10^{-13}
†	جنون	7.17×10^{-14}

Fig.8. System's output of the sample "حنون" from simplified Arabic font.

عباس

System output		$P(O\|\lambda)$
	عباس	7.12×10^{-11}
†	عياش	3.72×10^{-11}
‡	غياش	7.51×10^{-12}
‡	غباس	2.98×10^{-12}
	عباس	5.84×10^{-12}

Fig.9. System's output of the sample "عباس" from Arabic traditional font.

تماما

System output		$P(O\|\lambda)$
†	تماما	1.99×10^{-10}
‡†	بماما	1.25×10^{-10}
†	نماما	3.12×10^{-11}
†	تماما	1.96×10^{-11}
†	تمام	4.34×10^{-13}

Fig.10. System's output of the sample "تماما" from simplified Arabic font.

Figs.5, 6, and 7 show the system outputs of three samples from each font; simplified Arabic, Thuluth, and Arabic traditional respectively. Here the system output shows the same word more than once, which means the same word was recognized by a different path through the HMM. From time to time, the system throws up a sequence, not included in the lexicon, as shown in the tables.

* † In Figs.5–10, means not in the lexicon, and ‡ means not a valid Arabic word.

Removing these sequences from the solution list or otherwise considering their successors could enhance the overall recognition rate of the system. Although it is difficult to predict in advance which untrained words the HMM will recognize, it is found that a number of words recognized by the HMM were neither in the training set nor in the lexicon. Example word is shown in Figs.8 and 9.

The HMM is not always able to list the correct words among the best five paths. An example case may be seen in Fig.10 in which a dot is missing owing to a problem with thinning.

5 Conclusion

A method for the recognition of printed Arabic characters using Hidden Markov Models has been presented. Also, the problem in recognizing printed Arabic characters, and the important research techniques of Arabic character recognition have been discussed in this paper. The problem arises because of the segmentation stage, which is in fact similar to the segmentation of cursive script in many languages.

A single HMM model has been used for recognizing in which each word in the lexicon is presented by a single path through the model. This makes the system less sensitive to distortion and variation.

Finally, applying this technique to handwriting script seems very attractive as a future work.

References

[1] Al-Badr B, Mohmoud S. Survey and bibliography of Arabic text recognition. *Signal Processing*, 1995, 4: 49–77.

[2] Tolba M, Shaddad E. On the automatic recognition of printed Arabic characters. In *Proc. the Int. Conf. System, Man and Cybernetic*, USA, 1990, Vol.4–7, pp.496–498.

[3] Amin A, Alsadoun H. Hand printed Arabic character recognition system. In *Proc. the 12th Int. Conf. Pattern Recognition*, 1994, Vol.2, pp.536–539.

[4] Alherbish J, Ammar R. High-performance Arabic character recognition. *The Journal of System and Software*, 1998, 44: 53–71.

[5] Amin A. OCR of Arabic character. In *Proc. the 4th Int. Conf. Pattern Recognition*, 1988, pp.616–625.

[6] Khella F. Analysis of hexagonally sampled images with application to Arabic cursive text recognition [Dissertation]. University of Bradford, England, 1992.

[7] Al-Badr B, Haralick M A. Recognition without segmentation: Using mathematical morphology to recognize printed Arabic. In *Proc. the 13th National Computer Conference*, Saudi Arabia, 1992, pp.813–829.

[8] Makhoul J, Schwartz S. What is an HMM? *IEEE Spectrum*, 1997, pp.46–48.

[9] Rabiner L, Juang B. An introduction to hidden Markov Models. *IEEE ASSP Magazine*, Jan., 1986, pp.4–16.

[10] Rabiner L. A tutorial on Hidden Markov Models and selected applications in speech recognition. In *Proc. the IEEE*, February, 1989, Vol.77, pp.257–286.

[11] Bunke H, Roth M, Schukattalamazzini E. Off-line cursive handwriting recognition using Hidden Markov Models. *Pattern Recognition*, 1995, 28(9): 1399–1413.

[12] Chen M, Kundu A, Zhou J. Off-line handwriting word recognition using an HMM type stochastic network. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1994, 16(5): 481–496.

[13] Gray R M. Vector quantization. *IEEE ASSP Magazine*, 1989, 1: 4–29.

Abbas H. Hassin received the B.Sc. and M.Sc. degrees in computer science from Basrah University, Iraq in 1990 and 1995, respectively. From 1992–1995 he was a director of Database Information and Electronic Survey Division at College of Engineering in Basrah University. From 1995–2001 he was a lecturer at College of Science at the same university. Currently, he is a Ph.D. candidate in computer science at Harbin Institute of Technology (HIT), China. His research interests are Pattern Recognition and Character Recognition.

Xiang-Long Tang is the director of the T&R Division of pattern recognition and intelligent system at Harbin Institute of Technology, deputy director of Wearable Computer Engineering Center. He is a member of Council of Pattern Recognition and Machine Intelligence, Chinese Automation Society. His research field includes character recognition, image recognition, Chinese information processing, palm computer, biometrics computing. Prof. Tang has led 8 projects sponsored by the state, ministry/province and international cooperation. He has won a third prize of National Science and Technology Progress, a second prize, a third prize of Ministry level for Science and Technology Progress.

Jia-Feng Liu is an associate professor of Harbin Institute of Technology. He received B.S. and Ph.D. degrees from Harbin Institute of Technology (HIT) in 1990 and 1996 respectively. His research interests include pattern recognition, image processing and artificial intelligence.

Wei Zhao received her M.S. degree in pattern recognition and intelligence system from HIT in 2000. She worked on signature verification and word recognition in HIT from 1998. She is now a lecturer and a Ph.D. candidate at School of Computer Science and Technology, HIT. Her current research interests include pattern recognition, continuous character recognition and Chinese information process.