2012 International Congress on Informatics, Environment, Energy and Applications-IEEA 2012
IPCSIT vol.38 (2012) © (2012) IACSIT Press, Singapore

Farsi Handwritten Word Recognition Using Discrete HMM and Self-Organizing Feature Map

Behrouz.Vaseghi[1] and Somayeh.Hashemi [2]

[1,2] Young Researchers club,Abhar Branch,Islamic Azad University,Abhar,Iran

Abstract. A holistic system for the recognition of handwritten Farsi/Arabic words using right-left discrete hidden Markov models (HMM) and Kohonen self-organizing vector quantization(SOFM/VQ) for reading city names in postal addresses is presented. Pre-processing techniques including binarization, noise removal and besieged in a circumferential rectangular are described. Each word image is scanned form right to left by a sliding window and from each window 20 features (4*5) are extracted. The neighbourhood information preserved in the self-organizing feature map (SOFM) was used for smoothing the observation probability distributions of trained HMMs.

A separate HMM is trained by Baum Welch algorithm for each city name. A test image is recognized by finding the best match (likelihood) between the image and all of the HMM words models using forward algorithm. Experimental results show the advantages of using SOFM/HMM recognizer engine instead of conventional discrete HMM.

Keywords: Farsi Handwritten; HMM; pattrern recognition ; Self-Organizing Feature Map;

1. Introduction

During the past decade, pattern recognition community has achieved very remarkable progress in the field of handwritten word recognition. Many paper dealing with applications of handwritten word recognition to automatic reading of postal addresses, bank checks and forms (invoices, coupons, revenue documents etc.) have been published [1-5]. However, most of the works dealt with the recognition of Latin and Chinese scripts. However progress in Arabic script recognition has been slow mainly due to the special characteristics of Arabic scripts. Arabic text is inherently cursive both in handwritten and printed forms and is written horizontally from right to left. The reader is referred to [6-9] for further details on Arabic script characteristics and also the state of the art of Arabic character recognition. Farsi writing, which this paper addresses, is very similar to Arabic in terms of strokes and structure. The only difference is that Farsi has four more characters than Arabic in its character set Therefore, a Farsi word recognizer can also be used for Arabic word recognition. This paper presents a Farsi handwritten word recognition system based on discrete hidden Markov model and using a self-organization feature map as the vector quantization. This method is suitable for limited vocabulary applications such as postal address reading.

The work described has been carried out on a database of name of 198 cities of Iran.

2. The word recognition system

The proposed system is designed for reading the city names from address filed. The lexicon size is limited (198 city names) or can be pruned by using additional information like Zip codes. The block diagram of the system is illustrated in figure 1. An image of postal envelope is captured using a scanner with 300-dpi resolution and 256 gray levels. Then the name of city is extracted from the image and assigned an appropriate label between 1 to 198. Our database consists of 17000 word image of the cities in Iran.

[1] Tel:+98-912-740-7267
E-mail address: Behrouz.Vaseghi@Gmail.com
[2] Tel:+98-912-342-1632
E-mail address: somayyeh.Hashemi@Gmail.com

2.1. Pre-processing

The pre-processing consists of the following steps:

- Binarization: The gray level image of a word is binarized at a threshold determined by modified version of maximum entropy sum and entropic correlation methods.[10]
- Noise removal: The binarized image often has spurious segments which are removed by a morphological closing operation followed by a morphological opening operation both with a 3x3 window as the structure element.
- To Surround: For decrease of the memory volume and increase the speed of the processing the binarized image is surrounded in a circumferential rectangular (figure 2).

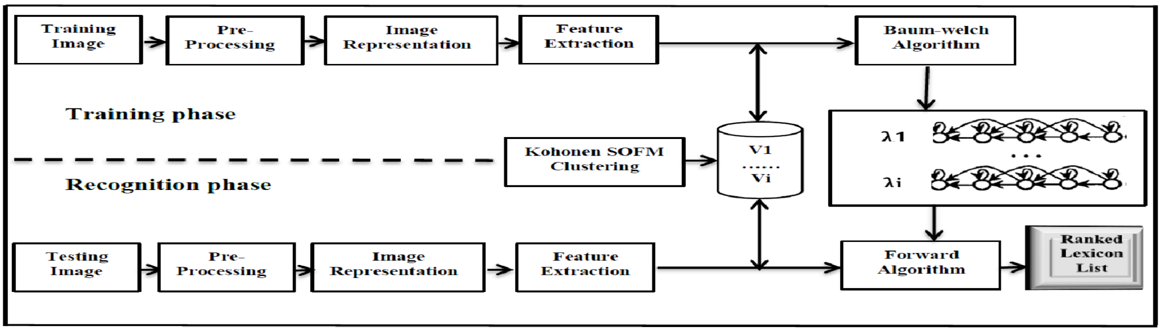


Fig.1. An overview of handwritten word recognition system

2.2. Frame generation

In this phase, the word image is converted to an appropriate sequential form suitable for HMM recognition engine. The area of the image is divided into a set of vertical fixed-width frames (strips) from right to left. The width of a frame is set to approximately twice of the average stroke width of the word image(r) and there is a 50% overlap between two consecutive frames. Then each frame is divided horizontally into five zones with equal height (M) as shown in figure 3. Therefore $zone_{i,j}$ shows the j_th zone from i_th frame.

2.3. Feature Extraction

In this stage from each zone of the frame of the image 4 features were extracted:

$$f_1(i,j) = [\sum_{y=1}^{2r}\sum_{X=1}^{M} zone_{i,j}(X,Y)] \times \frac{100}{M} \quad (1)$$

$$f_2(i,j) = [\sum_{Y=1}^{2r}\frac{1}{M}\sum_{X=1}^{M} x \times zone_{i,j}(X,Y)] \times \frac{100}{M} \quad (2)$$

$$f_3(i,j) = [\sum_{Y=1}^{2r}\frac{1}{M^2}\sum_{X=1}^{M} x^2 \times zone_{i,j}(X,Y)] \times \frac{100}{penwidth} \quad (3)$$

The fourth feature is defined as run position in the zone, therefore the fourth feature is:

$$f_4(i,j) = [start \quad \& \quad end \_ run \_ position] \times \frac{100}{M} \quad (4)$$

In this way, each frame is represented as a 20-dimensional feature vector as shown below:

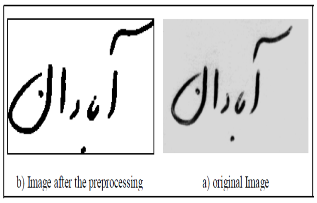

b) Image after the preprocessing a) original Image

Fig.2.An example image during reprocessing

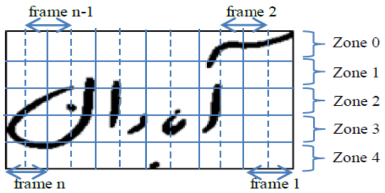
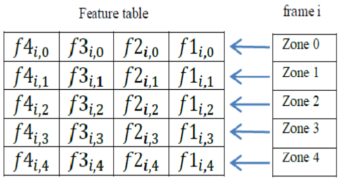


Fig.3.the final word image before feature extraction



$f4_{i,0}$	$f3_{i,0}$	$f2_{i,0}$	$f1_{i,0}$	←	Zone 0
$f4_{i,1}$	$f3_{i,1}$	$f2_{i,1}$	$f1_{i,1}$	←	Zone 1
$f4_{i,2}$	$f3_{i,2}$	$f2_{i,2}$	$f1_{i,2}$	←	Zone 2
$f4_{i,3}$	$f3_{i,3}$	$f2_{i,3}$	$f1_{i,3}$	←	Zone 3
$f4_{i,4}$	$f3_{i,4}$	$f2_{i,4}$	$f1_{i,4}$	←	Zone 4

Fig4.20-dimensional feature vector of frame i

3. Clustering and quantization

SOFM Clustering**:** The feature space must be quantized into a set of code word vectors in order to limit the number of observation symbols in discrete hidden Markov model training. The Kohonen self-organization feature map was used to construct the code book vectors. The extracted feature vectors from more than 400,000 word image frames (strips) are used as the input data file to the Kohonen SOFM clustering program (SOM PAK available via anonymous FTP [11]). The parameters used in SOFM clustering are shown in table 1.

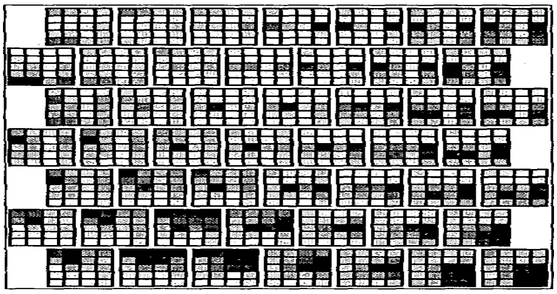


Fig.5. the SOFM represents codebook vectors.

Table 1. Parameter values used in SOM PAK program

Parameters	**Value**
Map topology	7x7 Hexagonal
Neighbourhood function	Bubble
Initial learning rate in first phase	0.1
Initial learning rate in second phase	0.05
Initial neighbourhood radius in first phase	10
Initial neighbourhood radius in second phase	3
Learning rate function	Inverse-time type

Figure 5 shows the best map with the minimum quantization error rate in 10 trials. The weight vectors of the map can be used as the code words of the codebook. Now, a feature vector of image frame can be assigned to the position index (Row, Column) of a node in the map which represents the closest code word according to the Euclidean distance. Therefore, each image word after this phase will be represented by a sequence of 2-dimensional code word positions in the map.

It is obvious from figure 5 that the SOFM will preserve the neighbourhood property of feature vectors across the map. This neighbourhood information is used for smoothing the trained HMM parameters, which will be explained in the following section.

4. Hidden Markov Model

The hidden Markov model is a double stochastic process, which can efficiently model the generation of sequential data [12]. HMMs have been successfully used in speech and handwriting recognition. There are two different approaches to model sequential data by HMM. In the model discriminant approach, a separate HMM is used for each class of patterns, while in the path discriminant approach, only one HMM models all of the pattern classes and different paths in the model distinguish one pattern class from the others. A model discriminant discrete HMM was used as the recognizer engine that is suitable for this limited-vocabulary application. Therefore, each city class (ω_c) is modelled by a single right-left HMM and a word is represented as a sequence of T observations (coordinates of the SOFM codebook), $O=\{o=(k,l\},1<t<T,1<k,l<7$ .An HMM, λ_c, is defined by the following parameters [12-13]:

The number of states (N) which is set for each class is proportional to the average numbers of frames in training samples in that class. The individual states are denoted as

$$S=\{S_1,S_2,..,S_N\} \tag{5}$$

and the state at time t as q_t .

The number of distinct observation symbols per state (M), which is equal to the SOFM codebook size (7x7). The individual symbols are denoted as $V=\{v_{k,l}\} \quad 1<k,l<7$

The state transition probability distribution:

$$A=\{a_{ij}\} \tag{6}$$

That $a_{ij}=p[q_{t+1}/q_t=s_i]\ ,1\le i,j\le N$ (7)

And $a_{ij}=0 \quad if(j\prec i)or(j\succ i+\Delta)$ (8)

The maximum number of forward jumps in each state (Δ) is chosen experimentally to be between 2 and 4 for each class during training.

The observation symbol probability distribution :

$$B=\{b_j(k,l)\} \tag{9}$$

That

$$b_j(k,l)=p[v_{k,l}att|q_t=s_j]\ ,1\le j\le N,1\le k,l\le M \tag{10}$$

The initial state distribution :

$$\Pi=\{\pi_i\} \quad ,1\le i\le N \tag{11}$$

That

$$\pi_i=p[q_i=s_i]=\begin{cases}0,i\ne 1\\ 1,i=1\end{cases} \tag{12}$$

The last state distribution :

$$\Gamma=\{\gamma_i\} \quad ,1\le i\le N \tag{13}$$

That

$$\gamma_i=p[q_T=s_i]=\begin{cases}0,i\ne N\\ 1,i=N\end{cases} \tag{14}$$

Prior to recognition, each HMM is trained independently by Baum-Welch algorithm [12-13] to maximize the probability of the observation sequences (obtained from training data set).

It is well known that if sufficient training data is not provided, HMM' parameters, especially the observation symbol probabilities, are usually poorly estimated. Consequently, the recognition rate becomes significantly degraded with even a slight variation in the testing data. This fact is clearly revealed in the experimental result shown in table2. An appropriate smoothing of the estimated observation probability can overcome this problem without the need for more training data. This was achieved by using the neighbourhood information preserved in the SOFM codebook [14]. After training all of the HMMs by Baum-Welch algorithm, the value of each observation probability in each state will be raised by adding a weighted-sum of the probabilities of its neighbouring nodes- in the self-organization map as follows:

$$b_j^{new}(k,l) = b_j^{old}(k,l) + \sum_{(k,l)\neq(p,q)} W_{(k,l),(p,q)} b_j^{old}(p,q) \qquad (15)$$

where the weighting coefficient $W_{(k,l),(p,q)}$ function of the distance between two nodes (p,q) and (k,l) in the map:

$$W_{(k,l),(p,q)} = sf.c^{(d_{(k,l),(p,q)}-1)} \qquad (16)$$

and c is a constant chosen to be equal to 0.5. The smoothing factor (Sf) controls the degree of smoothing, and $d_{(k,l),(p,q)}$ is the hexagonal distance between two nodes with the-coordinates (k,l) and (p,q) in the code book map.

The probability that o has been generated by each word model, ($P(o|\lambda_c),1<c<198$) ,was computed by forward algorithm as follows: [12-13]

The forward variable for a given word sample K is calculated as:

$$\alpha_t^{(K)}(j) = \begin{cases} \pi_j.b_j.(o_t^k), t=1 \\ \left[\sum_{i=1}^{N} \alpha_{t-1}^{(k)}(i).a_{ij}\right] b_j(O_t^{(k)})\square \quad \le t \le T_k \end{cases} ,1 \le j \le N \qquad (17)$$

Similarly, the backward variable for given word sample K is calculated as:

$$\beta_t^{(k)}(j) = \begin{cases} \gamma_j, t = T_K \\ \left[\sum_{i=1}^{N} a_{ji} b_j(O_t^{(k)}) \beta_{t+1}^{(k)}(i)\right], t = T_K - 1,..., 1 \end{cases} ,1 \le j \le N \qquad (18)$$

Finally the observation probability is calculated as:

$$P_K = P(O^{(K)} | \lambda_C) = \sum \alpha_{T_K}^{(K)}(i).\gamma_i \qquad (19)$$

and a sorted list of candidate classes were obtained.

5. Experimental results and Conclusion

A database consisting of about 17000 images of 198 city names of Iran is used for developing Farsi handwritten word recognition. After applying pre-processing steps including binarization, noise removal and besieged in a circumferential rectangular each word image is scanned from right to left by a sliding window and from each window 20 features is extracted. A codebook is constructed using SOFM clustering method from a pool of about 400,000 feature vectors extracted from word images. By using this codebook each word is represented as a sequence of membership vectors. For each city name a separate right-left HMM is trained by Baum-Welch algorithm under different conditions such as:

- Initializing parameters with random or equal values.
- Different topology parameters (the number of states (N), the connectivity of states (A)).

Then, each word image in the test data set was represented as a sequence of T observations, $O = \{o_t\}$. The probability that O has been generated by each word model $P(o|\lambda_c),1<c<198$, was computed by forward algorithm.

For computing the recognition rate three distinct sets (A, B, C) are predefined in the database usable for training and testing system. We use two of them for training and one set for testing and the mean of recognition rate in each condition is considered as recognition rate of the system in that condition.

The performance of the word recognition system is illustrated in table 2 by a top-n recognition rate measure (the percentage of samples that the true class is among the first n positions in the candidate list).

Table 2. Recognition result before smoothing HMM parameters.

Top-n	1	2	5	1	2

				0	0
Recognition Rate	33.21	45.47	70.52	87.21	93.98

As previously mentioned, due to the problem of insufficient training data, the recognition rate is unacceptable. The above procedure was repeated after smoothing the HMMs with a different smoothing parameter (Sf) and the recognition results are shown in table 3. Improvement is quite evident.

Table 3. Recognition result after smoothing HMM parameters

Top-n	1	2	5	10	20
Rec. Rate ($sf = 10^{-1}$)	59.98	74.58	85.41	89.95	94.35
Rec.Rate ($sf = 10^{-2}$)	66.42	79.12	87.27	92.85	96.09
Rec. Rate ($sf = 10^{-3}$)	63.84	75.12	84.95	91.58	95.81
Rec. Rate ($sf = 10^{-4}$)	59.77	72.77	85.71	92.02	95.98

Table 4. Comparisons to other word recognition system in the literature

Method	Lexicon size	Top 1	Top 2	Top 20
SCHMM,[6]	69	89.97	92.17	95.49
Proposed system	198	66.42	79.12	96.09
DHMM+smooting[2]	198	65.05	76.09	95
DHMM+VQ [1]	198	80.75	86. 10	94.46

From table 3, it is seen that the top-1 recognition rate increased significantly from 33.21% without smoothing, to 66.42% with a smoothing factor equal to 0.01.

In table 4 the performance of the propose word recognition system is compared with the other word recognition systems.

Examination our results and comparison with the other works, show the performance of propose system and features extracted in Farsi / Arabic handwritten word recognition. In this work is used from a simple pre-processing and basic concept of discrete HMM .

6. References

[1] Vaseghi.B., Alirezaee.Sh., "Off-line Farsi/Arabic Handwritten word recognition using vector quantization and hidden markov model," Proceedings of the 12th IEEE International Multitopic Conference, 978-1-4244-2824-3/08/$25.00

[2] Dehghan M., Faez K., Ahmadi M. and Shridhar M., “Handwritten Farsi (Arabic) word recognition: a holistic approach using discrete HMM”, Pattern Recognition, vol.34, no 5, pp. 1057-1065, 2001

[3] Chen M. Y., Kundu A., Srihari S. N., "Variable Duration Hidden Markov and Morphological Segmentation for Handwritten Word Recognition," IEEE Transactions on Image Processing, Vol. 4, No. 12, PP. 1675-1688, 1995.

[4] Kim G., Govindaraju V., "A Lexicon Driven Approach to Handwritten Word Recognition for Real-Time Applications," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 4, PP. 366-379, 1997.

[5] Guillevic D., Suen C. Y., "HMM-KNN Word Recognition Engine for Bank Cheque Processing,"Proceedings of International Conference on Pattern Recognition, Vol. 2, PP. 1526-1529, Brisebane, Ausrtalia, August 1998.

[6] Mario P., Maergner V. , “HMM Based Approach for Handwritten Arabic Word Recognition Using the IFN/ENIT Database”Proceedings of the seventh international conference on document analysis and recognition IEEE 2003

[7] Amin A., "Off-line Arabic Character Recognition: The State of The Art," Pattern Recognition, Vol 31, No. 5, PP. 517-530, 1998.

[8] Hamdani.M., El Abed.H., " Combining Multiple HMMs Using On-line and Off-line Features for Off-line Arabic Handwriting Recognition," 978-0-7695-3725-2/09 $25.00 © 2009 IEEE

[9] Al-Badr, B., Mahmoud S. A., "Survey and Bibliography of Arabic Optical Text Recognition," Signal Processing, Vol. 41, PP. 49-77, 1995.

[10] SahooP.,WilkinsC.,YeagerJ.,“Threshold Selection Renyi´s entropy” pattern recognition Vol. 30, No. 1, PP.71-84 1997

[11] Kohonen T., Hynninen J., Kangas J., Laaksonen J., "SOM_PAK: The Selp-Organizing Map Program Package, Version 3.1" Helsinki University of Technology, Finland, 1995.

[12] Mikael Nilsson “First order Hidden Markov Model Theory and implementation issues”Research Report Department of Signal Processing Blekinge institute of Tecnology Sweden. February, 20

[13] Rabiner L. R., "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," Proceedings of The IEEE, Vol. 77, No. 2, PP. 257-286, 1989.

[14] Zhao Z., Rowden C.G., "Use of Kohonen Self-Organising Feature Maps for HMM Parameter Smoothing in Speech Recognition," DEE Proceedings, Part F, Vol. 139, No. 6, PP. 385-390, 1992.2739