JKAU: Eng. Sci., **Vol. 19** No. 1, pp: 43-60 (2008 A.D. / 1429 A.H.)

Comparative Study of ANN and HMM to Arabic Digits Recognition Systems

Yousef Ajami Alotaibi

College of Computer & Information Sciences, King Saud University,
P.O. Box 57168, Riyadh 11574, Saudi Arabia
yalotaibi@ccis.ksu.edu.sa

Abstract. Arabic language is a Semitic language that has many differences when compared to Latin languages such as English. One of these differences is how to pronounce the ten digits, zero through nine. All Arabic digits are polysyllabic (except digit zero which is a monosyllabic) words and most of them contain Arabic unique phonemes, namely, pharyngeal and emphatic subset. In a previous paper the researcher designed an Artificial Neural Networks (ANN) based Arabic digits recognition system. In this paper we continued the research by designing Hidden Markov Model (HMM) based system that was designed and tested with automatic Arabic digits recognition. The old system was isolated whole word speech recognizer, but the current one was an isolated word phoneme based recognizer. Both systems were implemented both as a multi-speaker (*i.e.*, the same set of speakers were used in both the training and testing phases) mode and speaker-independent (*i.e.*, speakers used for training are different from those used for testing) mode. The main aim of this paper was to compare, analyze, and discuss the outcomes of these two recognition systems. The ANN based recognition system achieved 99.5% correct digit recognition in the case of multi-speaker mode, and 94.5% in the case of speaker-independent mode. On the other hand, the HMM based recognition system achieved 98.1% correct digit recognition in the case of multi-speaker mode, and 94.8% in the case of speaker-independent mode.

Keywords: Arabic, Digits, ASR, HMM, ANN.

1. Introduction

1.1 *Arabic Language*

Arabic is a Semitic language, and it is one of the oldest languages in the world. It is the fifth widely used language nowadays. Arabic is the first language in the Arab world, *i.e.*, Saudi Arabia, Jordan, Oman, Yemen, Egypt, Syria, Lebanon, etc. Arabic alphabets are used in several languages, such as Persian, and Urdu[1]. Arabic phonemes contain two distinctive classes, which are named pharyngeal and emphatic phonemes. These two classes can be found only in Semitic languages like Arabic and Hebrew[2, 3].

Modern Standard Arabic (MSA) has basically 34 phonemes, of which six are basic vowels, and 28 are consonant[2]. A phoneme is the smallest element of speech units that indicates a difference in meaning, word, or sentence. Arabic language has fewer vowels than English language. It has three long and three short vowels, while American English has at least twelve vowels[4]. The allowed syllables in Arabic language are: CV, CVC, and CVCC where V indicates a (long or short) vowel while C indicates a consonant. Arabic utterances can only start with a consonant[2].

The researches on Arabic language are mainly concentrated on MSA, which is used throughout the media, courtrooms and academic institutions of the Arabic countries. Previous work on developing Automatic Speech Recognition (ASR) was dedicated to dialectal and colloquial Arabic within the 1997 NIST benchmark evaluations, and more recently on the recognition of conversational, dialectal speech, as is reported in[5].

The development of accurate ASR systems is faced with two major issues. The first problem is related to diacritization because diacritic symbols refer to vowel phonemes in the designated words. Arabic texts are almost never fully diacritized: it means that the short strokes placed above or below the consonant, indicating the vowel following this consonant, are usually absent. This limits the availability of Arabic ASR training material. The lack of this information leads to many similar word forms, and then, decreases predictability in the language model. The second problem is related to the morphological complexity since Arabic has a rich potential of word forms which increases the out-vocabulary rate[6, 7].

1.2 *Spoken Digits Recognition*

Automatic recognition of spoken digits is one of the challenging tasks in the field of computer ASR. Spoken digits recognition process is needed in many applications that need numbers as input, such as telephone dialing using speech, addresses, airline reservation, automatic directory to retrieve or send information, etc. Arabic digits zero to nine are polysyllabic words except the first one, zero, which is a monosyllable word [2] as shown in Table 1.

Table 1. Arabic digits[11].

Dig it	Arabic writing	Pronunciation	Syllables	No. of syllables
1	واحد	wâ-hĕd	CV-CVC	2
2	اثنين	‘aâth-nāyn	CVC-CVC	2
3	ثلاثه	thâ-lă- thâh	CV-CV-CVC	3
4	أربعه	‘aâr-bâ-‘aâh	CVC-CV-	3
5	خمسه	khâm-sâh	CVC-CVC	2
6	سته	sĕt-tâh	CVC-CVC	2
7	سبعه	sûb-‘aâh	CVC-CVC	2
8	ثمانيه	thâ-mă-nyĕh	CV-CV-CVC	3
9	تسعه	tĕs-âh	CVC-CVC	2
0	صفر	sĕfr	CVCC	1

Arabic language had limited number of research efforts compared to other languages such as English and Japanese. A few researches have been conducted on the Arabic digits recognition. In 1985, Hagos[8] and Abdullah[9] separately reported Arabic digit recognizers. Hagos designed a speaker-independent Arabic digits recognizer that used template matching for input utterances. His system is based on the LPC parameters for feature extraction and log likelihood ratio for similarity measurements. Abdullah developed another Arabic digits recognizer that used positive-slope and zero-crossing duration as the feature extraction algorithm. He reported 97% accuracy rate. Both systems mentioned above are isolated-word recognizers in which template matching is used. Al-Otaibi[10] developed an automatic Arabic vowel recognition system. Isolated Arabic vowels and isolated Arabic word recognition systems were implemented. He studied the syllabic nature of the Arabic language in terms of syllable types, syllable structures, and primary stress rules.

1.3 *Neural Networks*

Artificial Neural Networks (ANNs) have been investigated for many years for the desire of achieving human-like performance in the field of ASR. These models are composed of many nonlinear computational elements operating parallel in patterns similar to the biological neural networks[12]. ANN has been used extensively in ASR field during the past two decades. The most beneficial characteristics of ANNs for solving ASR problem are the fault tolerance and nonlinear property[13].

ANN models are distinguished by the network topology, node characteristics, and training or learning rules. One of the important models of the neural networks is the multilayer perceptrons (MLPs), which are feed-forward networks with zero, one, or more hidden layers of nodes between the input and output nodes[12]. The capabilities of the MLP stem from the nonlinearities used with its nodes. Any MLP network must consists of one input layer (not computational, but source nodes), one output layer (computational nodes), and zero or more hidden layers (computational nodes) depending on the network sophistication and the application requirements[13].

Many Arabic ASRs were designed using ANN techniques[11- 23]. In the first research a spoken Arabic digits recognizer was designed to investigate the process of automatic recognition process. The system was operated in two different modes, multi-speaker mode and speaker-independent mode. The overall system performance was 99.47% in the first mode and 96.46% in the second mode.

1.4 *Hidden Morkov Models*

ASR systems based on the Hidden Markov Model (HMM) started to gain popularity in the mid 1980's[14]. HMM is a well-known and widely used statistical method for characterizing the spectral features of speech frame. The underlying assumption of the HMM is that the speech signal can be well characterized as a parametric random process, and the parameters of the stochastic process can be predicted in a precise, well-defined manner. The HMM method provides a natural and highly reliable way of recognizing speech for a wide range of applications[15, 16]. The Hidden Markov Model Toolkit (HTK)[17] is a portable toolkit for building and manipulating HMM models. It is mainly used for designing, testing,

and implementing ASR and its related research tasks. The author and his research colleagues conducted many research using HMM to recognize, analyze, and investigate the spoken Arabic digits and alphabets as shown in Ref. [24- 29].

1.5 *Problem Definition and Goals*

The goal of this paper is to design an HMM based Arabic digit recognition system and evaluate its accuracy in two different modes, namely multi-speaker and speaker independent modes. This system was, then, compared with a similar past system[11] that was an ANN based one. To compare two systems regarding the system design approach of the system we have to keep similar training and testing data, extracted features, and other parameters. In this case we can study in an effective manner the effect of changing system type. The comparison included the overall system performance and the individual digit accuracy for both mentioned modes.

2. Experimental Framework

In this section the two different system approaches for ASR are presented in detail. The used data sets for training and testing for both modes are the same. In addition to that the common parameter values that are common in both systems were fixed to same values.

2.1 *ANN System Overview*[11]

An ASR based on neural networks was developed to carry out the goals of this research. This system was partitioned into several modules according to their functionality as shown in Fig. 1. First is the digital signal processing front-end module, whose functions are speech acquisition through a microphone, filtering, and sampling. A band-pass filter with cut-off frequencies 100Hz and 4.8 KHz was used to filter speech signal before processing. The sampling rate was set to 10 KHz with 16-bit resolution for all recorded speech tokens.

A manual endpoint detection method was, also, used to separate speech from silent portions of the signal. It also detects the beginning and the end points of the spoken word (digit)[18]. Linear predictive coding

(LPC) techniques were computed for sequential frames 64 points (6.4 ms) apart. In each case, a 256-point Hamming window was used to select the data points to be analyzed[19]. Linear predictive coding module calculates ten mel-frequency cepstrum coefficients (MFCCs), with LPC order (p=10), for each frame in the spoken utterance, thus 11 MFCC coefficients is extracted from each frame. For MFCC computations, 20 triangular band-pass filters were considered in feature extraction subsystem as in Ref. [20].

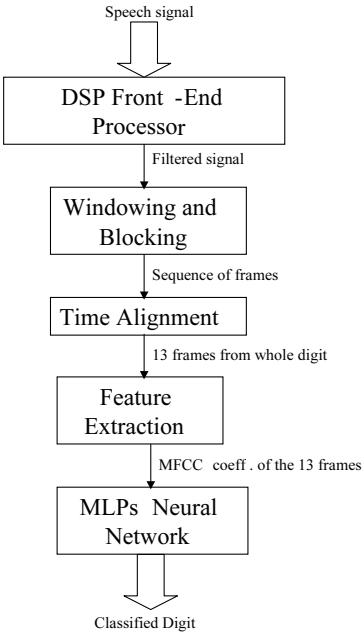

Fig. 1. ANN Based System Block Diagram[11].

A fully connected feed-forward multilayer perceptron (MLP) network was used to recognize the unknown spoken digit. All MLP neurons used logistic non-linearities and the back-propagation training algorithm[13]. The network consists of 143 nodes in the input layer (source nodes). The number of nodes in this layer depends on the number of MFCC coefficients for every frame and the number of considered frames in the whole token that is currently applied on the input layer. Number of considered frames is 13 (11 MFCC coefficients × 13 frames=143) depending on our simple and effective time-alignment algorithm[11].

The MLP network contains two hidden layers with 40 nodes in the first hidden layer and 15 nodes in the second hidden layer. The output layer consists of 10 neurons. Each neuron in the output layer should be *on* or *off* depending on the applied digit on the input layer. For the normal and intended situation, only one node should be *on* while all others should give an *off* state if the applied utterance is one of the ten Arabic digits, otherwise, all neurons should output *off* state.

2.2 *HMM System Overview*

An ASR based on HMM was developed to carry out the goals of this research. This system was partitioned into three modules according to their functionality as shown in Fig. 2. First is the training module, whose function is to create the knowledge about the speech and language to be used in the system. Second is the HMM models bank, whose function is to store and organize the system knowledge gained by the first module. Finally is the recognition module whose function is to try to figure out what is the (meaning) of the input speech given in the testing phase. This was done with the aid of the HMM models mentioned above.

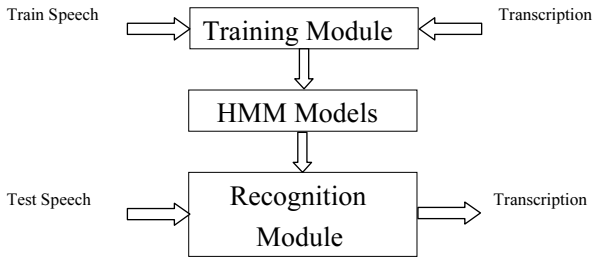

Fig. 2. HMM based system block diagram.

The parameters of the system are 10 KHz sampling rate with 16 bit sample resolution, 25 millisecond Hamming window duration with step size of 10 millisecond, MFCC coefficients with 22 as the length of cepstral leftering and 26 filter bank channels, 12 as the number of MFCC coefficients, and 0.95 as the pre-emphasis coefficients as can be seen in Table 2.

Table 2. System parameters[11].

Parameter	Value
Sampling rate	10 khz, 16 bits
Database	Isolated 10 Arabic digits
Speakers	17
Repititions	10
Filter cut-off frequencies	100 hz and 4.8 khz
Preemphased	1-0.95 z^{-1}
Window type and size	Hamming, 256
Window step size	64
LPC order	10

The second system proposed in this paper was implemented using HMM technique with the help of HTK tools. The speech ASR was designed initially as phoneme level recognizer with 3-state, continuous, left-to-right, no skip HMM models. The system was designed by considering all 37 MSA monophones as given by Language Data Consortium (LDC) catalog[21]. The silence (sil) model was also included in the model set. In a later step, the short pause (sp) was created from and tied to the silence model. Since most digits consisted of more than two phonemes, context-dependent triphone models were created from the monophone models mentioned above. Before this the monophones models were initialized and trained by the training data explained above. This was done by more than one iteration and repeated again for triphones models. The training phase step before the last is to align and tie the model by using a decision tree method. The last step in training phase was to re-estimate HMM parameters using Baum-Welch algorithm[15] three times.

2.3 *Database*

An in-house database was created from all ten Arabic digits. A number of 17 individual male Arabic native speakers were asked to utter all digits ten times. Hence, the database consists of 10 repetitions of every digit produced by each speaker, totaling of 1,700 tokens. All samples for a given speaker were recorded in one session. During the recording session, each utterance was played back to ensure that the entire digit was included in the recorded signal. All the 1,700 tokens were used for training and testing phases depending on system run mode.

We have in this research two modes, namely the multi-speaker mode and the speaker-independent mode. Table 2 shows some of the system parameters. This database was used in both systems in the same way for both modes.

3. Results

3.1 *Multi-Speaker Mode*

In the multi-speaker mode, the first and second repetitions of each digit that were uttered by all speakers were used for the training phase. Thus, the total tokens considered for training is 340 (17 speakers × 2 repetitions × 10 digits). For testing mode, all the 1,700 tokens were used in recognition phase (testing mode). This implies that the training data set is a subset of the testing data set. This data setting was applied for both the ANN based and HMM based systems.

Table 3 shows the accuracy of the ANN based system for digits individually in addition to the system overall accuracy. Depending on testing database set, the system must try to recognize 170 samples for every digit where the total number of tokens is 1,700. The overall system performance was 99.47%, which is reasonably high. The system failed in recognizing only 9 tokens out of the 1,700 total tokens. Digits 1, 5 6, 7, and 9 got 100% recognition rate, on the other hand, the worst performance was encountered with digits 4 and 8 where the performances was the same and it is equal to 98.24% (three tokens were miss-recognized in each case). In addition to that, our time-alignment algorithm is very simple and straightforward.

With the same procedure applied to the HMM based system, Table 4 shows the accuracy for the digits individually in addition to the system overall accuracy. The overall system performance was 98.06%, which is reasonably high. The system failed in recognizing only 25 tokens out of the 1,700 total tokens. Digits 1, 2, 4, 6, and 7 got 100% recognition rate, on the other hand, the worst performance was encountered with digit 0 where the performances was 88.24%. This digit was confused mainly with digit 7 where 19 tokens of digit 0 were recognized as digit 7. Even though the database size is small (only the ten spoken Arabic digits), the system showed an unexpected high performance due to the variability in how to pronounce Arabic digits and the fact that we considered multi-

speaker mode in contrast to speaker-dependent mode (one speaker only trains and uses the system).

Table 3. ANN confusion matrix (multi-speaker mode)[11].

	one	two	three	four	five	six	seven	eight	nine	zero	Acc. (%)
one	170	–	–	–	–	–	–	–	–	–	100
two	1	169	–	–	–	–	–	–	–	–	99.41
three	–	–	169	–	–	1	–	–	–	–	99.41
four	–	1	–	167	–	–	2	–	–	–	98.24
five	–	–	–	–	170	–	–	–	–	–	100
six	–	–	–	–	–	170	–	–	–	–	100
seven	–	–	–	–	–	–	170	–	–	–	100
eight	1	–	1	–	–	–	–	167	–	1	98.24
nine	–	–	–	–	–	–	–	–	170	–	100
zero	1	–	–	–	–	–	–	–	–	169	99.41
Average											99.47

Table 4. HMM confusion matrix (multi-speaker mode).

	one	**two**	**three**	**four**	**five**	**six**	**seven**	**eight**	**nine**	**zero**	**Acc. (%)**
one	170	–	–	–	–	–	–	–	–	–	100
two	–	170	–	–	–	–	–	–	–	–	100
three	–	1	161	8	–	–	–	–	–	–	94.71
four	–	–	–	170	–	–	–	–	–	–	100
five	–	–	1	–	169	–	–	–	–	–	99.41
six	–	–	–	–	–	170	–	–	–	–	100
seven	–	–	–	–	–	–	170	–	–	–	100
eight	–	–	–	–	1	–	1	168	–	–	98.82
nine	–	–	–	1	–	–	–	–	169	–	99.41
zero	–	–	–	–	1	–	19	–	–	150	88.24
Average											98.06

3.2 *Speaker-Independent Mode*

In speaker-independent mode, on the other hand, four speakers (speakers one through four) were used for the training phase purpose. The total samples dedicated for this phase is 400 (4 speakers × 10 repetitions × 10 digits). The testing set consists of utterances of speakers

5 trough 17 with ten repetitions and ten digits. A total token prepared for the testing phase is 1,300 tokens (13 speakers × 10 digits × 10 repetitions). This data setting was applied for both the ANN based and HMM based systems.

In contrast to the multi-speaker mode, speaker-independent mode of the ANN based was used in configuring the system and the performance is shown in Table 5. The total tokens tested by the system are 1,300 (130 for every digit). The overall system accuracy is 94.26% with total of 72 miss-recognized tokens. The worst performance was found in the case of digit 1 (with accuracy equal to 86.92%); and the best performance was encountered in the case of digit 9 (with accuracy equal to 100%), and digit 3 (with accuracy equal to 97.96%).

Table 5. ANN confusion matrix (speaker-independent mode)[11].

	one	**two**	**three**	**four**	**five**	**six**	**seven**	**eight**	**nine**	**zero**	**Acc. (%)**
one	113	1	7	6	–	–	2	–	–	1	86.92
two	2	126	–	–	–	–	–	1	1	–	96.92
three	–	–	127	1	–	–	–	–	–	2	97.69
four	2	1	–	123	1	–	2	–	1	–	94.62
five	–	–	–	4	125	–	–	–	–	1	96.15
six	–	–	3	–	–	126	–	–	–	1	96.92
seven	–	–	–	7	–	2	12 1	–	–	–	93.08
eight	–	–	9	–	–	–	–	118	–	3	90.77
nine	–	–	–	–	–	–	–	–	130	–	100
zero	–	–	–	3	–	2	–	–	6	119	91.54
Average											94.46

By switching to the other mode, speaker-independent mode of the HMM based was used in configuring the system, it gave a lower accuracy rate as shown in Table 6. The overall system accuracy is 94.77% with total of 68 miss-recognized tokens. The worst performance was found in the case of digit 0 (with accuracy equal to 83.08%); and the best performance was encountered in the case of digits 1 and 7 (with accuracy equal to 100%), and digit 2 (with accuracy equal to 99.23%). In general, for speaker-independent mode, this overall performance is acceptable if we keep in mind the complication of the recognition task and the high similarity between Arabic digits.

Table 6. HMM cnfusion matrix (speaker-independent mode).

	one	two	three	four	five	six	seven	eight	nine	zero	Acc. (%)
one	130	–	–	–	–	–	–	–	–	–	100
two	1	129	–	–	–	–	–	–	–	–	99.23
three	–	1	127	–	–	–	2	–	–	–	97.69
four	4	–	–	114	3	–	2	7	–	–	87.69
five	2	–	1	–	124	–	1	2	–	–	95.38
six	4	–	–	–	–	126	–	–	–	–	96.92
seven	–	–	–	–	–	–	130	–	–	–	100
eight	1	–	–	–	–	–	1	128	–	–	98.46
nine	4	–	4	2	3	–	–	1	116	–	89.23
zero	5	–	–	–	–	–	17	–	–	108	83.08
Average											94.77

Figures 3 & 4 and Tables 7 & 8 depicted extra information about the performance of both systems and both modes. The conclusion is the ANN approach is better than HMM approach in designing Arabic digit recognition systems and this may be true for all simple recognizers with less than 50 word vocabulary size. This conclusion was supported by a number of facts as follows. First, in all modes the ANN based system gave a better or equal an overall performance compared to HMM based system. Second, the performance of individual digits is, in general, better in ANN system. Third, the complexity and cost of ANN based system is much less than that of HMM based system. Finally, it is easier to maintain and modify ANN based system.

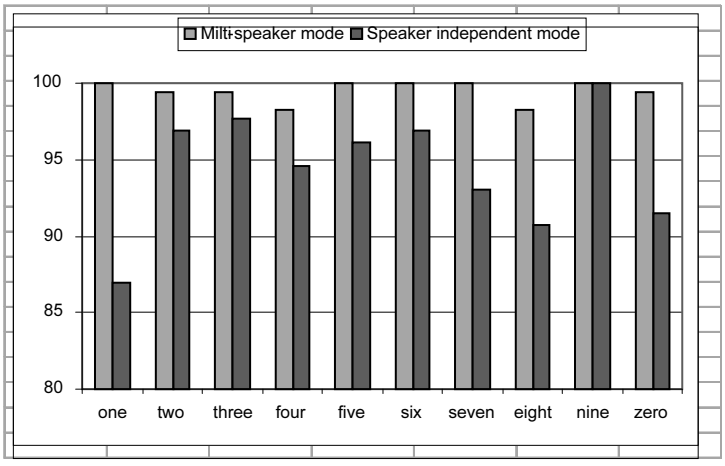

Fig. 3. ANN accuracy rate for individual Arabic digits for both modes[11].

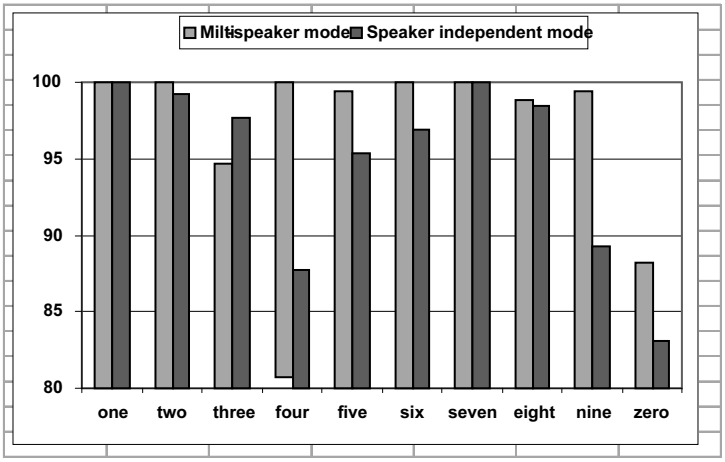

Fig. 4. HMM accuracy rate for individual Arabic digits for both modes.

Table 7. ANN Digits that were picked in case of miss-recognition for both modes and all digits[11].

Digit	Confused with digit(s)	
	Multi-speaker Mode	Speaker Independent Mode
1	–	2, 3, 4, 7, 0
2	1	1, 8, 9
3	6	4, 0
4	2, 7	1, 2, 5, 7, 9
5	–	4, 0
6	–	3, 0
7	–	4, 6
8	1, 3, 0	3, 0
9	–	–
0	1	4, 6, 9

Table 8. HMM Digits that were picked in case of miss-recognition for both modes and all digits

Digit	Confused with digit(s)	
	Multi-speaker Mode	Speaker Independent Mode
1	–	–
2	–	1
3	2, 4	2, 7
4	–	1, 5, 7, 8
5	3	1, 3, 5, 7, 8
6	–	1
7	–	–
8	5, 7	1, 7
9	–	1, 3, 4, 5, 8
0	5, 7	1, 7

3.3 *Discussion*

Regarding the multi-speaker mode, we noticed that the overall performance of ANN based system was higher than that of the HMM based system by almost 1.5%. Also we noticed that the number digits that got 100% accuracy in ANN based system was more than that in HMM based system. Depending on related tables and figures, we concluded that ANN based system is better in performance than HMM based digit recognition system. The justification for this might be that in the case of HMM approach we are using a very sophisticated and big approach to solve a very simple and small problem. It is a small problem because the number of vocabulary is only 10 and this implied a very simple speech recognition problem that can be solved by pattern comparison approach by means of ANN. In other words, HMM based systems are useless in simple speech recognition problems.

On the other hand, for the speaker-independent mode, the overall performance for both ANN and HMM based systems was almost identical. In speaker-independent mode the recognition problem turned to a relatively more difficult one compared to multi-speaker mode. Justification for this equally performances might be that ANN based system could not overcome this relative difficulty so that its overall performance dropped sharply but with HMM based system this difficulty is in fact “piece of cake”, thus this system gave higher performance.

4. Conclusion

Two spoken Arabic digit recognizers were designed to investigate the process of automatic recognition process. The first system was an ANN based while the second one was an HMM based. Using two different modes, namely multi-speaker mode and speaker independent mode, The ANN based system and HMM based system performances were compared. It has been found that in multi-speaker mode the performance of the ANN system was better than that of HMM based system. On the other hand, in speaker-independent mode the two performances were almost identical. Finally we concluded from this investigation that the ANN approach is better than HMM approach in designing Arabic digit recognition systems due to its simplicity of such recognizer.

References

[1] **Al-Zabibi**, **M.**, *An Acoustic-Phonetic Approach in Automatic Arabic Speech Recognition*, The British Library in Association with UMI, 1990.

[2] **Alkhouli**, **M.,** *Alaswaat Alaghawaiyah*, Daar Alfalah, Jordan 1990 (in Arabic).

[3] **Elshafei**, **M.,** Toward an Arabic Text-to -Speech System, *The Arabian Journal for Science and Engineering*, **16**(4B): 565-83, Oct. 1991.

[4] **Deller, J., Proakis, J.** and **H.Hansen**, **J.,** *Discrete-Time Processing of Speech Signal*, Macmillan, 1993.

[5] **Kirchhoff, K., Bilmes, J., Das, S., Duta, N., Egan, M., Gang J., Feng H., Henderson, J., Daben L., Noamany, M., Schone, P., Schwartz, R.** and **Vergyri, D.**, Novel approaches to Arabic speech recognition: report from the 2002 Johns-Hopkins Summer Workshop, *Proceedings of ICASSP 2003*, Vol. **1**, pp: 344-347, April 2003.

[6] **El-Imam**, **Y. A.,** An Unrestricted Vocabulary Arabic Speech Synthesis System, *IEEE Transactions on Acoustic, Speech, and Signal Processing*, **37**(12) Dec. :1829-45, 1989.

[7] **Omar**, **A.,** *Derasat Alaswat Aloghawi*, Aalam Alkutob, Eygpt, 1991 (in Arabic).

[8] **Hagos**, **E.**, *Implementation of an Isolated Word Recognition System*, UMI Dissertation Service, 1985.

[9] **Abdulah, W.** and **Abdul-Karim**, **M.,** Real-time Spoken Arabic Recognizer, *Int. J. Electronics*, **59**(5) : 645-648, 1984.

[10] **Al-Otaibi**, **A.,** *Speech Processing*, The British Library in Association with UMI, 1988.

[11] **Alotaibi**, **Y. A.,** High Performance Arabic Digits Recognizer Using Neural Networks, *The 2003 International Joint Conference on Neural Networks –IJCNN 2003*, Portland, Oregon, 2003.

[12] **Lippmann**, **R.,** *Review of Neural Networks for Speech Recognition*, Neural Computation, pp.1-38, MIT press, 1989.

[13] **Haykin**, **S.,** *Neural Networks: A Comprehensive Foundation*, Second Edition, Prentice Hall 1999.

[14] **Loizou, P.C.** and **Spanias**, **A.S.,** High-Performance Alphabet Recognition, *IEEE Trans. on Speech and Audio Processing*, **4**(6) Nov.: 430-445, 1996.

[15] **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, Feb. 1989.

[16] **Juang, B.** and **Rabiner**, **L.**, Hidden Markov Models for Speech Recognition, *Technometrics*, **33**(3) August, pp: 251-272, 1991.

[17] **Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V.** and **Woodland**, **P.,** *The HTK Book* (for HTK Version. 3.4), Cambridge University Engineering Department, 2006. http:///-htk.eng.cam.ac.uk/prot-doc/ktkbook.pdf.

[18] **Rabiner, L.** and **Samber**, **M.**, An Algorithm for Determining the Endpoints of Isolated Utterances, *The Bell System Technical Journal*, **54**(2): 297-315, 1975.

[19] **Nocerino, N., Soong, F., Rabiner, L.** and **Klatt**, **D.,** Comparative Study of Several Distortion Measures for Speech Recognition, *Speech Communication*, **4**: 317-31, 1985.

[20] **Davis, S.** and **Mermelstein**, **P.,** Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences, *IEEE Trans. on Acoustic, Speech, and Signal Processing*, **ASSP-28**(4) Aug, 1980.

[21] *Linguistic Data Consortium (LDC) Catalog Number LDC2002S02*, http://www.ldc.upenn.edu/, 2002.

[22] **Rabiner, L.** and **Wilpon**, **J.**, A Simplified, Robust Training Procedure for Speaker Trained Isolated Word Recognition Systems, *J. Acoustic Society of America*, **68**(5) November, 1980.

[23] **Alotaibi**, **Y.A.,** Spoken Arabic Digits Recognizer Using Recurrent Neural Networks, *International Symposium on Signal Processing and Information Technology- ISSPIT*, 195-199, Rome, Italy, 2004.

[24] **Alotaibi, Y.A., Abdullah-Al-Mamun, K.** and **Muhammad**, **G.**, Study on Unique Pharyngeal and Uvular Consonants in Foreign Accented Arabic, *INTERSPEECH 2008*, Brisbane, Australia, 22-26 September 2008 (accepted).

[25] **Alotaibi, Y.** and **Selouani**, **S.,** Experiments on Adaptation to Non-Native Arabic Accent in Automatic Speech Recognition, *Journal of Saudi Computer Society*, Applied Computing and Information, Accepted on Feb.27, 2008.

[26] **Alotaibi, Y.A., Selouani, S.** and **O'Shaughnessy**, **D.,** Experiments on Automatic Recognition of Nonnative Arabic Speech, *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2008, Article ID 679831, 9 pages, 2008. doi:10.1155/2008/679831.

[27] **Alotaibi, Y. A., Alghamdi, M.** and **Alotaiby**, **F.,** Using A Telephony Saudi Accented Arabic Corpus in Automatic Recognition of Spoken Arabic Digits, *4th International Symposium on Image/Video Communications over fixed and mobile networks (ISIVC '08)*, Bilbao - Spain, July 9-11th, 2008,

[28] **Alotaibi**, **Y. A.,** Investigating Spoken Arabic Digits in Speech Recognition Setting, *Information Sciences Journal*, **173**: 115-139 (2005).

[29] **Selouani, S.** and **Alotaibi**, **Y.A.**, Investigating Automatic Recognition of Non-Native Arabic Speech, *IEEE conference on Innovations in Information Technology*, pp: 451-455, Dubai, UAE, 2007.

دراسة مقارنه بين أداء الخلايا العصبية و نموذج ماركوف الخفي في أداء التعرف الآلي على الأرقام العربية المنطوقة

يوسف عجمي العتيبي

جامعة الملك سعود، ص. ب. ٥٧١٦٨ الرياض ١١٥٧٤

المملكة العربية السعودية

yalotaibi@ccis.ksu.edu.sa

المستخلص. اللغة العربية من اللغات السامية التي تختلف اختلافات كثيرة بالمقارنة إلى اللغات اللاتينية مثل اللغة الإنجليزية. ومن هذه الاختلافات، كيفية نطق الأرقام العربية العشرة من الصفر وحتى التسعة، حيث أنها جميعها متعددة المقاطع ماعدا الصفر، فإنها تحتوي على مقطع واحد فقط. كذلك تتميز اللغة العربية بوجود الأصوات الحلقية والمفخمة التي لا توجد في كثير من اللغات الأخرى. في هذا البحث يتم إكمال بحث سابق للباحث حيث اعتمد البحث السابق على خوارزم الخلايا العصبية الاصطناعية في التعرف آليا على الأرقام العربية المنطوقة، وفي هذا البحث يتم الاعتماد على خوارزم نموذج ماركوف الخفي والمقارنة في أداء الطريقتين لحل هذه المسألة. في البحث السابق تم اعتماد الكلمة كوحدة في المعالجة والتعرف، أما في هذا البحث فتم اعتماد الصوت (الفونيم) كوحدة بديلة، وفي جميع الحالات يتم التعامل مع الكلمة المنطوقة وهي في معزل عن الكلمات السابقة واللاحقة. في كلا البحثين تم اعتماد طريقتين في اختيار العينات الصوتية لمرحلة التدريب والاختبار. الطريقة الأولى هي المزج بين المتكلمين أنفسهم في التدريب والاختبار، أما الطريقة الثانية فهي جعل عينات التدريب منطوقة بمتحدثين غير الذين تحدثوا في عينات الاختبار.

والهدف من هذا البحث هو مقارنة وتحليل مناقشة أداء هاتين الخوارزميتين في عملية التعرف الآلي على الحروف العربية المنطوقة. في حل هذه المسألة حققت خوارزمية الخلايا العصبية الاصطناعية نسبة دقة ٩٩,٥٪ و ٩٤٪ بينما حققت خوارزمية نموذج ماركوف الخفي نسبة دقة ٩٨,١٪ و ٩٤,٨٪.