International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:7, No:5, 2013

Efficient DTW-Based Speech Recognition System for Isolated Words of Arabic Language

Khalid A. Darabkh, Ala F. Khalifeh, Baraa A. Bathech, and Saed W. Sabah

***Abstract*—**Despite the fact that Arabic language is currently one of the most common languages worldwide, there has been only a little research on Arabic speech recognition relative to other languages such as English and Japanese. Generally, digital speech processing and voice recognition algorithms are of special importance for designing efficient, accurate, as well as fast automatic speech recognition systems. However, the speech recognition process carried out in this paper is divided into three stages as follows: firstly, the signal is preprocessed to reduce noise effects. After that, the signal is digitized and hearingized. Consequently, the voice activity regions are segmented using voice activity detection (VAD) algorithm. Secondly, features are extracted from the speech signal using Mel-frequency cepstral coefficients (MFCC) algorithm. Moreover, delta and acceleration (delta-delta) coefficients have been added for the reason of improving the recognition accuracy. Finally, each test word's features are compared to the training database using dynamic time warping (DTW) algorithm. Utilizing the best set up made for all affected parameters to the aforementioned techniques, the proposed system achieved a recognition rate of about 98.5% which outperformed other HMM and ANN-based approaches available in the literature.

***Keywords*—**Arabic speech recognition, MFCC, DTW, VAD.

I. Introduction

THE Arabic language is considered nowadays as the fifth widely used language [1] as there are more than 200 million people speak this language. Unfortunately, the research efforts are still limited in comparison with other languages such as English and Japanese as far as the automatic speech recognition is concerned. However, it is noteworthy to mention that the Arabic digits (one-nine) are polysyllabic words while zero is a monosyllable word [2]. On the other hand, Arabic phonemes can be found of two categories, namely, pharyngeal and emphatic phonemes. These categories are found in only Semitic languages such as Hebrew [2-3]. However, automatic speech recognition (ASR) has received a great deal of attention by many researchers for a decade which basically allows a computer to recognize spoken words recorded by its microphone. Speech recognition is used in a wide area of applications include interfacing with deaf people, home automation, healthcare, robotics, and much more. Actually, various approaches were adopted for speech recognition which are mainly found in three categories, Template-based such as dynamic time warping (DTW), neural network-based such as artificial neural networks (ANNs), and statistics-based such as hidden Markov models (HMMs).

In this paper, we propose an efficient DTW-based speech recognition system for isolated words of Arabic language. A brief summary of our system is as follows: A preprocessing is made for not only noise reduction, but also normalization. Moreover, speech/non-speech regions of the voice signal are detected using voice activity detection (VAD) algorithm. In addition, segmenting the detected speech regions into manageable and well-defined segments for the purpose of facilitating the upcoming tasks has been considered. Hence, the segmentation of speech can be divided into two types; the first one is called "Lexical", which divides a sentence into separate words, while the other type is called "Phonetic", which is based on dividing each word into phones. After that, the Mel-frequency cepstral coefficients (MFCC) approach was adopted due to its robustness and effectiveness compared to other well-known feature extraction approaches like linear predictive coding (LPC) [4-5]. Moreover, delta and acceleration coefficients were added to MFCC for the sake of improving the accuracy of Arabic speech recognizer. Finally, DTW were used as a pattern matching algorithm due to its speed and efficiency in detecting similar patterns [6-7]. Many experiments were conducted to find the best parameters required to achieve the best efficient Arabic speech recognizer.

Unlike other languages, Arabic language is characterized by having tremendous dialectal variety, diacritic text material, as well as morphological complexity which all in turn challenge the researchers in proposing highly accurate Arabic recognition system. In [8], a morphology-based language model was investigated for the use in a speech recognition system for conversational Arabic. In [9], the authors investigated the discrepancies between dialectal and formal Arabic in a speech recognition system utilizing morphology-based language model, automatic vowel restoration, as well as the integration of out of corpus language model. In [10], the authors reported the feasibility of using the automatic diacritizing Arabic text in acoustic model training for ASR. In [11], the authors attempted to use CMU (Carnegie Mellon University) Sphinx speech recognition system, which is one of the most robust speech recognizers in English, to develop an extension useful for Arabic language. However, more relevant research articles will be discussed and compared with our

K. A. Darabkh is with the Department of Computer Engineering, The University of Jordan, Amman 11942, Jordan (phone: +962-77-9103900; e-mail: k.darabkeh@ju.edu.jo).

Ala F. Khalifeh is with the Department of Communication Engineering, German Jordan University, Amman 11180, Jordan (phone: +962 6 429 4112; e-mail: ala.khalifeh@gju.edu.jo).

Baraa A. Bathech and Saed W. Sabah are with the Department of Computer Engineering, The University of Jordan, Amman 11942, Jordan.

work in the results and discussions Section I.

The rest of the paper is divided into three further sections. Section II describes the proposed system. Section III presents our experimental results, observations, and discussions. Finally, Section V concludes our work.

II. The Proposed System

The proposed Arabic speech recognition system consists of many stages which are summarized as follows:

A. Database Collection

There is a need for a feature database that includes stored spoken words in Arabic for pattern matching process explained later. We have built a feature database of 100 utterances (Arabic words and digits) for testing purposes produced by 30 speakers (19 males and 11 females) who were asked to record each word three times. An important point to mention is that words stored in our database were recorded in a normal home environment with a sampling rate of 8 KHz and 16 bit depth. This process is illustrated in Algorithm#1. Further details about what are mentioned in Algorithm#1 can be found while discussing the subsequent subsections.

Algorithm#1: **Pseudo code for creating the features database**

```
//IN:
PATH = "Recorded Speech Path";
File;
//OUT:
DB = Database File;
Begin
    Generating Database;
    for File = 1:number of files in PATH do
        Read the File;
        Apply VAD;
        Extract Features;
        Save Features in DB;
    Endfor
End
```

B. Preprocessing

This stage aims to enhance some signal characteristics in order to achieve more accurate results through canceling disturbances that may affect the quality of the recorded speech. This stage can be divided into two steps as follows.

Step#1: Pre-emphasis

At this step, high frequency contents of the input signal are emphasized in order to flatten the signal's spectrum. In our paper, the pre-emphasizer is represented by a first order FIR filter [12], which can be described according to:

$$H(z) = 1 - 0.97z^{-1} \quad (1)$$

where, z refers to discrete Fourier transform of the speech signal. However, the effect of applying this filter for a sample word in Arabic is shown in Fig. 1 whereas we can see that the high amplitude pulses in the signal that adversely affect the accuracy of stored word features in extraction stage are significantly reduced.

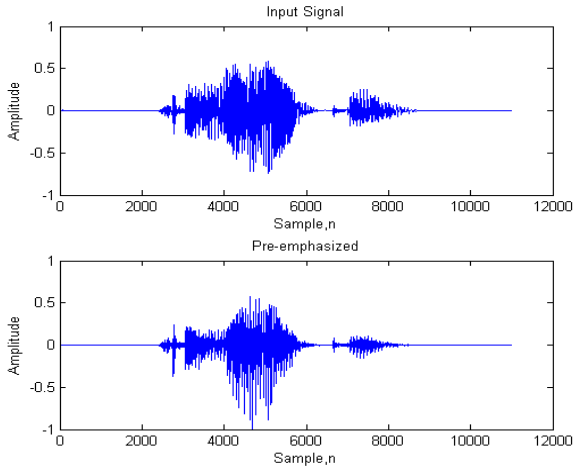

Fig. 1 The word "عليكم" after the pre-emphasis

Step#2: Hearingization

Speakers usually defer in speaking loudness, and since different microphones defer in their sensitivity to speech, hearingization is included in our experiments which can be found as [13-15]:

$$S_1(n) = \frac{x_{pre-emphasis}(n) - Mean\left(x_{pre-emphasis}(n)\right)}{Max\left(\left|x(n) - Mean(x_{pre-emphasis}(n))\right|\right)} \quad (2)$$

The hearingized version of the signal is depicted in Fig. 2.

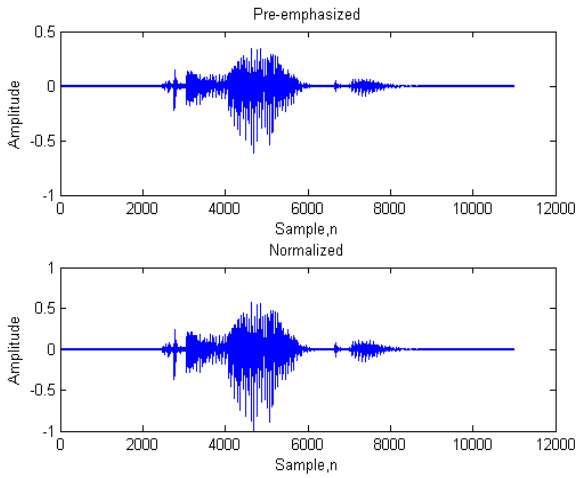

Fig. 2 The word "عليكم" after the hearingization

C. Voice Activity Detection (VAD)

Generally, one of the major problems that affect the efficiency of a speech recognizer is detecting the start and end points of voice activity. However, short-term power and zero-crossing rate are commonly used parameters for distinguishing speech/non-speech regions [15]. Hence, this stage can be divided into the following steps:

Step#1: Framing

The speech signal is segmented into non-overlapped frames where each has a width of 20ms. Non-overlapping frames are used to reduce the number of times needed to check for voice activity. Consequently, the overall processing time of this stage is reduced.

Step#2: Short-term power and zero-crossing rate

It is worth mentioning that the short-term power is significantly increased in speech regions. However, it can be calculated according to [14]:

$$P_{S_1}(m) = \frac{1}{L}\sum_{n=m-L+1}^{m} {S_1}^2(n) \tag{3}$$

where m, L, and n refer to frame number, frame length, frame index respectively. On the other hand, zero-crossing rates tend to have larger values in non-speech regions. This gives a good indication of speech existence. In fact, it can be calculated according to [15]:

$$Z_{S_1}(m) = \frac{1}{L}\sum_{n=m-L+1}^{m} \frac{\left|\text{sgn}(S_1(n)) - \text{sgn}(S_1(n-1))\right|}{2} \tag{4}$$

where,

$$\text{sgn}(S_1(n)) = \begin{cases} +1: & \text{S}_1(\text{n}) \geq 0 \\ -1: & \text{S}_1(\text{n}) < 0 \end{cases} \tag{5}$$

Step#3: Speech Indicator

The aforementioned parameters are combined in the following formula in order to provide a more comfortable approach which can be used to calculate a threshold value based on its mean and standard deviation [12, 15]:

$$W_{s1}(m) = P_{s1}(m)\left(1 - Z_{s1}(m)\right)F \tag{6}$$

where F is a constant which is used to avoid having small values. However, to initiate this function, we use the following activation function (AF):

$$AF_W = M_W + cSD_W \tag{7}$$

where, M_W refers to the mean of $W_{s1}(m)$, SD_W refers to the standard deviation of $W_{s1}(m)$, and c is a constant which should be fine-tuned since it depends on signal characteristics. Accordingly, the voice activity detection function can be found as:

$$VAD(m) = \begin{cases} 1, & \text{W}_{s1}(\text{m}) \geq AF_W \\ 0, & \text{Otherwise} \end{cases} \tag{8}$$

The result of this segmentation process is shown in Fig. 3. The output signal after doing VAD is $x_1(n)$ where it is simply $s_1(n)$ when VAD (n) is on. The stage of implementing VAD is described in *Algorithm#2*.

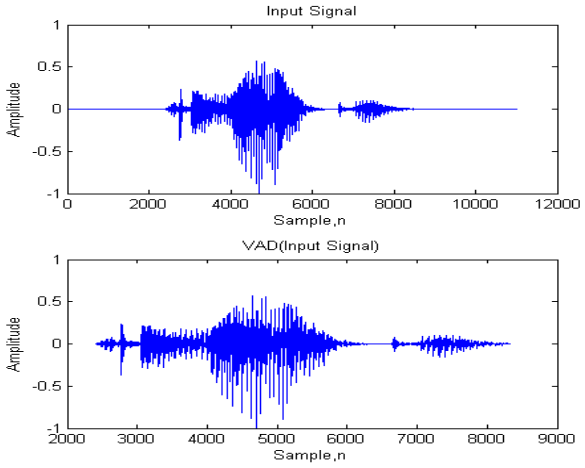

Fig. 3 The word "عليكم" as detected by VAD

Algorithm#2: **Pseudo code for implementing VAD algorithm**

```
//IN:
FrameLength = 160;
Overlap = 0%;
Signal;
//OUT:
VAD_Signal;
Begin
    Framed = Framing(Signal, FrameLength, Overlap);
    P = Power(Framed);
    Z = ZeroCrossing(Framed);
    W = P * (1-Z);
    M = Mean(W);
    S = Std(W);
   AF = M +  c * S;
   VAD_Signal = (W>AF) * Signal;
End
```

D. Feature Extraction

The feature extraction phase consists of the following steps:

Step#1: Framing

In our experiments, the voice signal ($x_1(n)$) is then broken up into J frames of P samples for each one with an overlapping ratio of 36.5%, so that adjacent frames are separated by T samples (where $T<P$). The chosen values for P and T are 240

samples and 87 samples, respectively which were so appropriate. Hence, the output signal contains *J* vectors of length *P*, which corresponds to $x_1(p;j)$, where *p=0,1,2,...,P-1* and *j=0,1,2,...,J-1*.

Step#2: Hamming Window

Applying hamming window, to the output signal discussed in step#1 (framed signal), helps in reducing discontinuity at both ends of each frame and this can be done utilizing the following formula [15]:

$$Ham(p) = 0.54 - 0.46\cos\frac{2\pi p}{P-1}, \quad 0 \le p \le P-1 \qquad (9)$$

where *i* refers to the sample index and *N* indicates the length of a frame (in samples). By applying *Ham (p)* to $x_1(p;j)$ for all frames, then $x_2(p;j)$, which refers to the windowed signal, is easily found.

Step#3: Fast Fourier Transform

To study the characteristics of the speech signal in frequency domain, we use *N-point* FFT to convert the windowed signal, resulting from step#2, from time domain to frequency domain. Note that the frame length here is a power of 2 ($N=2^i$), hence the output signal is $X_2(n;j)$.

Step#4: Mel Filter Bank

According to the fact that human perception of voice frequencies is nonlinear (i.e., human hearing is less sensitive at higher frequencies, roughly > 1000 Hz), a Mel-scale is used so that for each tone with a frequency *F* measured in Hz, a subjective pitch is measured on a Mel-scale according to following formula [12-14]:

$$F_{mel} = 2595\log_{10}\left(1+\frac{F_{Hz}}{700}\right) \qquad (10)$$

After finding the magnitude of $X_2(n;j)$ and using the Mel scale filter bank (which consists of 30 triangular-band-pass filters which have an equal spacing before 1 kHz and logarithmic scale after 1 kHz), the Mel spectrum coefficients are found as the summation of the filtered results as the following:

$$Mel_v = \sum_{n=0}^{N-1}\left|X_2(n;j)\right|TF_v^{mel}(n) \qquad (11)$$

where $TF_v^{mel}(n)$ is the n[th] triangular filter.

Step#5: Inverse Discrete Cosine Transform

To this end, we should return back to time domain. The best technique to do this while achieving highly uncorrelated features is the inverse discrete cosine transform (IDCT) as found in Equation (12). Before finding that, we compute first the logarithm of the magnitude of the output of Mel-filter bank since logarithm compresses dynamic range of values whereas humans are less sensitive to slight differences in amplitude at high amplitudes than low amplitudes.

$$IC(p;j) = \sum_{i=0}^{P-1}\lambda_i \log(Mel_i)\cos\left(\frac{\pi(2p+1)i}{2P}\right), \quad p = 0, 1, 2, ..., P-1 \qquad (12)$$

where, $\lambda_0 = \sqrt{1/P}$ and $\lambda_i = \sqrt{2/P}$, $1 \le i \le P-1$

Step#6: Liftering

To extract the vocal tract cepstrum, it is good to use liftering which is mainly a filtering in the spectrum domain. The simplest way to do that is to drop some of the cepstrum coefficient at the end. However, the most popular lifter that gives very promising recognition result is [15]:

$$l(p) = \begin{cases} 1+\frac{Y-1}{2}\sin(\frac{\pi p}{Y-1}), & p = 0,1, ..., Y-1 \\ 0, \text{ Otherwise} \end{cases} \qquad (13)$$

In our experiments, the best value chosen for Y is $\frac{3}{4}P$. As a summary, we use the first 12 cepstral coefficients for each frame and ignore the rest which have the F0 spike. In our work, the MFCC consists of steps 1 though 6.

Step#7: Short-term Energy

The cepstral coefficients do not capture energy. Therefore, the log of signal energy is an interesting feature to increase the coefficients derived from Mel-cepstrum. In other words, for every frame, the following energy term is added:

$$E_j = \log\sum_{p=0}^{P-1}x_2^2(p;j) \qquad (14)$$

Step#8: Delta and Acceleration Coefficients

It is known that the speech signal is not constant. In other words, the slope of formants may change from stop burst to release [16]. Hence, it is worth adding these changes in the features (i.e., the slopes). These are called delta features and delta acceleration (delta-delta) features. The delta coefficients are computed using a linear regression formula given *2C+1* is the size of the regression window:

$$\Delta IC_l(m) = \frac{\sum_{i=1}^{C} i\left(IC_l(m+i) - IC_l(m-i)\right)}{2\sum_{i=1}^{C} i^2} \qquad (15)$$

where $IC_l(m)$ is the m[th] MFCC coefficient. As far as the delta-delta coefficients are concerned, they are found using linear regression of delta features. As a summary, we used 39-dimensional features as the following (12 MFCC, 1 energy

feature, 12 delta MFCC features, 12 delta-delta MFCC features, 1 delta energy feature, 1 delta-delta energy feature). The steps of obtaining the features' vectors are described in Fig. 4. On the other hand, *Algorithm#3* describes the steps for analyzing the signal and obtaining its features along with delta and delta-delta coefficients.

Algorithm#3: **Pseudo code for extracting the features vector**

```
//IN:
VAD_Signal;
FrameLength = 240;
Overlap = 36.5%;
K = 30;
HammingWindow;
//OUT:
MFCCs;
Features_Extraction;
Begin
    Framed = Framing(VAD_Signal);
    Hammed = Filter(Framed, HammingWindow);
    FFT_Signal = FFT(Hammed);
    MelCep = MelFilterBank(ABS(FFT_Signal)^2, K)
    Cep = Log(IDCT(MelCep));
    Lifted = Liftering(Cep,3:14);
    Energy = sum(Hammed^2);
    MFCC = [Energy:Cep];
    DeltaMFCC = Delta(MFCC);
    AccMFCC = Acceleration(MFCC);
    MFCCs = [MFCC:DeltaMFCC:AccMFCC]
End
```

E. Pattern Matching

Dynamic time warping algorithm, which is based on dynamic programming, is a technique that calculates the level of similarity between two time series in which any of them may be warped in a non-linear fashion by shrinking and stretching the time axis [5-7]. Fig. 5 shows the warp path between the test word and the trained word as a result of applying DTW algorithms on different utterances. As shown from this figure, two time series are warped to find the best alignment between them. The lines shown between the two time series connect points that have the same value but happened in different time instants. Moreover, if the compared time series were identical, all lines connected between them must be straight vertical. Importantly, the warp path represents the actual distance between the two time series which can be measured as the accumulative sum between each two identical points in the time series being under comparison [6, 16]. To this extent, we can summarize that any test word is segmented and its features are calculated and consequently compared with the whole database using DTW in order to find the word that has the nearest distance path to it. This is clearly described in *Algorithm#4*.

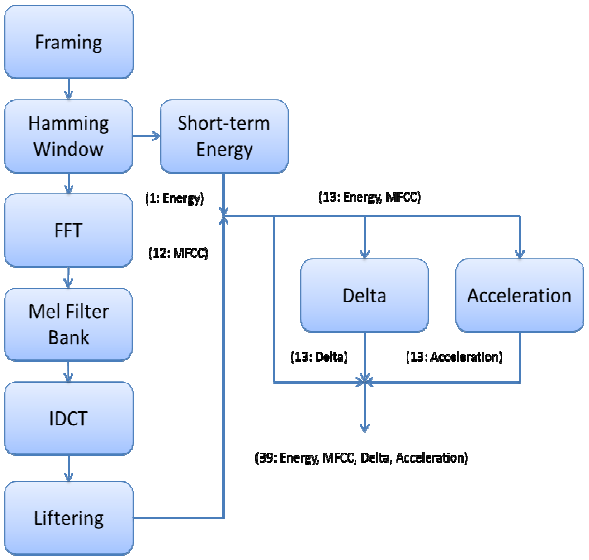

Fig. 4 Features extraction processing steps

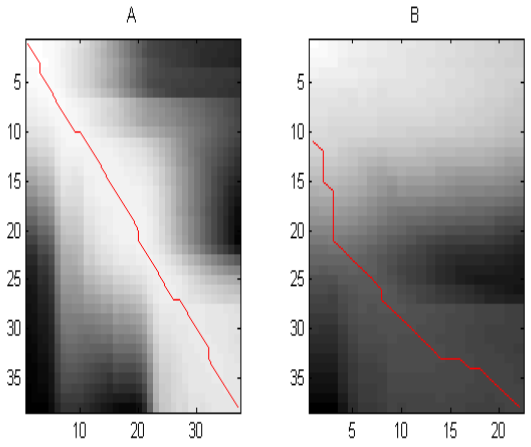

Fig. 5 A) The warp path between different utterances of the word "عليكم", B) The warp path between the words "عليكم" and "سبعة"

Algorithm#4: **Pseudo code for pattern matching**

```
//IN:
RecordedSpeech;
DB;
Begin
   VAD_Speech = VAD(RecordedSpeech);
   Features = FeatureExtraction(VAD_Speech);
   Cost = Infinity;
   For  N = 1:Length(DB)
     SM = SimilarityMatrix(Features, Feature(DB(N)));
     CurrentCost = dpfast(1-SM);
     If CurrentCost < Cost
           Cost = CurrentCost;
     Endif
   Endfor
End
```

III. Results and Discussions

A. Our Results

In order to evaluate the performance of the proposed system, recorded samples were split into training and testing sets whereas two thirds of them were used for training and the rest used for testing. It deserves mentioning that the minimum number of tests made to recognize an Arabic word is ten. Below is the formula which describes how the recognition rate of each word was calculated:

$$RR = \frac{Number\,\text{of correctly recongnized words}}{Number\,\text{of tested words}} * 100\% \qquad (16)$$

Table I describes the recognition rate for a sample of tested words in the database which we have previously recorded. For every test word, the recognition rates are calculated using three different combinations of features as shown in this table. The positive effect of employing VAD and MFCC on the recognition rate is clearly observed. Furthermore, adding delta and acceleration coefficients to the feature set improves the recognition rate significantly. This is to be expected since the delta coefficients find the first derivative of the feature set which adds an important parameter that reflects the changing of speech from a specific phoneme to the next one. It is noteworthy to mention that the first derivative may give noisy results. Thus, in our proposed system, it is combined with using the polynomial approximation approach. Consequently, the system's response for some tested words like "واحد" was improved. Additionally, by incorporating the polynomial approximation approach, it would be possible to calculate the second derivative of the features in order to give more important information to the feature set for the reason of improving the overall performance and accuracy of the system. Actually, this can be noticed when considering the word "كم" shown in Table I. To this extent, the fitting width ($2C+1$) adds a delay of C blocks to the system. The choice of the value of C is a tradeoff between good and accurate approximation and long delay. Hence, C was given a value of 3 in order to have a good accuracy with a relatively faster response.

B. Comparisons with Previous Work

There are interesting approaches, similar in target to our proposed system, done to improve the recognition rate of Arabic language. In [17], a heuristic method for Arabic speech recognition (ArSR), minimal eigenvalues algorithm, was used to find the most promising path through a tree of different samples of an uttered word. Furthermore, radial neural networks (RNN) approach was incorporated with this heuristic method to enhance the recognition rate. The recognition accuracies were about 86.45% and 95.82% for minimal eigenvalues algorithm and RNN, respectively. In [11], an Arabic speech recognition system was proposed using open source CMU Sphinx-4 and hidden Markov models. The obtained recognition accuracy was about 85.55%. In [18], comparisons were made between monophone, triphone, syllable, and word-based algorithms for recognizing Egyptian Arabic digits. Thirty-nine MFCC coefficients were extracted as features for every recorded voice in the database where they were used to train HMMs in which the system matches between the testing word and training database. The achieved recognition accuracies were about 90.75%, 92.24%, 93.43%, and 91.64% for monophone, triphone, syllable, and word-based recognition algorithms, respectively. In [19], an Arabic numeral recognition (ArNR) technique was proposed using vector quantization (VQ) and HMM whereas the LP cepstral coefficients were used. The recognition accuracy was about 91%.

In [20], HMM-based Arabic numeral recognition system was proposed using Wavelet cepstral coefficients and Mel frequency cepstral coefficients whereas the recognition accuracies for different numerals were about 61%-92% and 76%-92% for MFCC-based and Wavelet-based systems, respectively. In [21], other HMM-based approaches were proposed, based on LPC and MFCC, for Arabic numeral recognition and devised for field programmable gate arrays (FPGAs) whereas the recognition accuracy ranges from 91% to 96% for LBC-based recognition systems and 95% to 98% for MFCC-based recognition systems. In [5], a comparison of discrete hidden Markov model and DTW was made for recognizing isolated words in Arabic language. In DTW-based approach, they used 13 MFCC coefficients and also the same for delta and acceleration coefficients. A 256 point FFT was used to find the power spectrum to be used in an emulated filter-bank composed of 24 triangular weighting functions in Mel scale. After that, the natural logarithmic was applied to the 24 filter-bank. They measured the recognition rate for frames' overlap length of 512*256. The recognition accuracy was about 86% in clear environment using the characteristics of power (energy) and differential information (Δ and $\Delta\Delta$). In DHMM-based speech recognizer; five states were defined for each word whereas transitions between theses states are possible only in left to right direction with no states skipping. No more details were reported about these states and transitions. The achieved recognition accuracy was about 92%. The recognition rates obtained from aforementioned approaches are summarized below in Table II. All mentioned rates are obtained assuming clear environment. The significance of our proposed work is clearly noticed.

TABLE I
RECOGNITION RATES FOR DIFFERENT FEATURE SETS

Tested Word (Arabic Writing)	Transcription	English Writing	Approach#1: VAD+MFCC	Approach#2: VAD+MFCC+Δ	Approach#3: VAD+MFCC+Δ+ΔΔ
واحد	WAHID	ONE	85.7%	100%	100%
اثنان	ITHNAN	TWO	100%	100%	100%
ثلاثة	THALATHA	THREE	100%	100%	100%
أربعة	ARBAA	FOUR	100%	100%	100%
خمسة	KHAMSA	FIVE	100%	100%	100%
ستة	SITTA	SIX	85.7%	85.7%	85.7%
سبعة	SABAA	SEVEN	100%	100%	100%
ثمانية	THAMANIYA	EIGHT	100%	100%	100%
تسعة	TISAA	NINE	100%	100%	100%
عشرة	ASHRA	TEN	85.7%	100%	100%
السلام	ASSALAAMU	PEACE	100%	100%	100%
عليكم	ALAIKUM	UPON YOU	100%	100%	100%
كيف	KEEF	HOW	100%	100%	100%
حالك	HALAK	ARE YOU	85.7%	85.7%	85.7%
ما	MA	WHAT	100%	100%	100%
اسمك	ESMOK	YOUR NAME	100%	100%	100%
كم	KAM	HOW	85.7%	85.7%	100%
عمرك	OMROK	YOUR AGE	100%	100%	100%
مهنتك	MEHNATOK	YOUR OCCUPATION	100%	100%	100%

TABLE II
COMPARISONS WITH PREVIOUS WORK

Previous Work	*Recognition Rates*
Heuristic Method [17]	86.45%
Heuristic Method with RNN [17]	95.82%
ASR using CMUSphinx [11]	85.55%
Monophone-Based ArSR [18]	90.75%
Triphone-Based ArSR [18]	92.24%
Syllable-Based ArSR [18]	93.43%
Word-Based ArSR [18]	91.64%
VQ and HMM ArNR [19]	91%
MFCC-based ArNR [20]	61%-92%
Wavelet-based ArNR [20]	76%-92%
LBC-based FPGA ArNR [21]	91%-96%
MFCC-based FPGA ArNR [21]	95%-98%
DTW-Based ArSR [5]	86%
DHMM-Based ArSR [5]	92%
Our proposed recognition system	98.5%

IV. CONCLUSIONS

Finding efficient automatic speech recognition techniques for Arabic words is of great interest since the research efforts remain limited. In this work, the robustness of MFCC combined with DTW algorithm is clearly noticed. Moreover, the voice activity detector technique has a significant impact on system's performance. On the other hand, adding delta and delta-delta coefficients help in improving the overall recognition accuracy. Many experiments were conducted to choose the best parameters that maximize the improvements of Arabic speech recognition. Additionally, a noticeable speech recognition accuracy improvement is achieved when compared to other HMM and ANN-based approaches.

REFERENCES

[1] M. Al-Zabibi, "An Acoustic–Phonetic Approach in Automatic Arabic Speech Recognition," *The British Library in Association with UMI*, UK, 1990, http://hdl.handle.net/2134/6949.
[2] M. Alkhouli, "Alaswaat Alaghawaiyah," *Daar Alfalah*, Jordan, 1990 (in Arabic).
[3] M. Elshafei, "Toward an Arabic Text-to-Speech System," *The Arabian Journal for Science and Engineering*, vol. 16, no. 4B, pp. 565-83, October 1991.
[4] S.B. Davis, P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no.4, pp. 357–366, August 1980.
[5] Z. Hachkar, A. Farchi, B. Mounir, J. El Abbadi, "A Comparison of DHMM and DTW for Isolated Digits Recognition System of Arabic Language," *International Journal on Computer Science and Engineering*, vol.3, no.3, pp.1002-1008, March 2011.
[6] Lindasalwa Muda, Mumtaj Begam, I. Elamvazuthi, "Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW)", *Journal of Computing*, vol. 2, no. 3, pp. 138-143, March 2010.
[7] Stan Salvador, Philip Chan, "Toward Accurate Dynamic Time Warping in Linear Time and Space", *Intelligent Data Analysis Journal*, vol. 11, no. 5, pp. 561-580, October 2007.

[8] D. Vergyri, K. Kirchhoff, K. Duh, A. Stolcke, "Morphology-based language modeling for Arabic speech recognition", *In INTERSPEECH-2004*, pp. 2245-2248, 2004.
[9] K. Kirchho, J. Bilmes, J. Henderson, R. Schwartz, M. Noamany, P. Schone, G. Ji, S. Das, M. Egan, F. He, D. Vergyri, D. Liu, and N. Duta, "Novel Approaches to Arabic Speech Recognition," *Technical Report*, Johns-Hopkins University, 2002.
[10] D. Vergyri, K. Kirchhoff. "Automatic diacritization of Arabic for acoustic modeling in speech recognition", In Ali Farghaly and Karine Megerdoomian, editors, COLING 2004, *Computational Approaches to Arabic Scriptbased Languages*, pp. 66–73, Geneva, Switzerland, 2004.
[11] H. Satori, M. Harti, N. Chenfour, "Introduction to Arabic Speech Recognition Using CMUSphinx System," *Proceedings of Information and Communication Technologies International Symposium (ICTIS'07),* Fes, Morocco, pp. 139-115, July 2007.
[12] Lawrence Rabiner, Biing-Hwang Juang*, Fundamentals of speech recognition*, Upper Saddle River, New Jersey: Prentice Hall, USA, 1993
[13] X. Huang, A. Acero, and H.-W. Hon, *Spoken Language Processing*, Upper Saddle River, New Jersey: Prentice Hall, USA, 2001.
[14] B. Gold and N. Morgan, *Speech and Audio Signal Processing*, New York, New York: John Wiley and Sons, USA, 2000.
[15] Mikael Nilsson and Marcus Ejnarsson, "Speech Recognition using Hidden Markov Model (performance evaluation in noisy environment)", *Masters Thesis*, Department of Telecommunications and Signal Processing, Belkinge Institute of Technology, Ronneby, Sweden, March 2002.
[16] B.S. Jinjin Ye, "Speech Recognition Using Time Domain Features From Phase Space Reconstructions", *Masters Thesis*, Department of Electrical and Computer Engineering, Marquette University, Milwaukee, Wisconsin, May 2004.
[17] Khalid Saeed and Mohammad Nammous, Heuristic Method of Arabic Speech Recognition, *Bialystok University of Technology*, Poland, http://aragorn.pb.bialystok.pl/~zspinfo/
[18] Mohamed Mostafa Azmi, Hesham Tolba, Sherif Mahdy, Mervat Fashal, "Syllable-Based Automatic Arabic Speech Recognition", *Proceedings of WSEAS International conference of Signal Processing, Robotics and Automation (ISPRA' 08),* University of Cambridge, UK, pp. 246-250, February 2008.
[19] H. Bahi and M. Sellami, "Combination of Vector Quantization and Hidden Markov Models for Arabic Speech Recognition," *Proceedings of the ACS/IEEE International Conference on Computer Systems and Applications (AICCSA 2001)*, Beirut, Lebanon, pp: 96-100, June 2001.
[20] W. Alkhaldi, W. Fakhr, N. Hamdy, "Multi-Band Based Recognition of Spoken Arabic Numerals Using Wavelet Transform," *Proceedings of the 19th National Radio Science Conference (NRSC'01)*, Alexandria University, Alexandria, Egypt, March 19-21, 2002.
[21] F.A. Elmisery, A.H. Khalil, A.E. Salama, H.F. Hammed, "A FPGA Based HMM for a Discrete Arabic Speech Recognition System," *Proceedings of the 15th International Conference on Microelectronics (ICM 2003),* Cairo, Egypt, December 9-11, 2003.