Spoken Utterance Detection Using Dynamic Time Warping Method Along With a Hashing Technique

John Sahaya Rani Alex [#1], Nithya Venkatesan [*2]

[#]School of Electronics Engineering
VIT University- Chennai
Tamil Nadu, 600127
INDIA
[1]jsranialex@vit.ac.in

[*]School of Electrical Engineering
VIT University - Chennai
Tamil Nadu, 600127
INDIA
[2]nithya.v@vit.ac.in

***Abstract*- This paper presents a technique of searching a keyword in a spoken utterance using Dynamic Programming algorithm. This method is being revisited because of the evolution in computing power. The proposed methods present less computational complexity compared with the conventional Dynamic Time Warping (DTW) method. The proposed methods are tested with connected TIDIGIT data.**

Keywords— Keyword Spotting, Dynamic Time Warping (DTW), Speech Recognition, HTK Tool Kit, MFCC.

I. INTRODUCTION

Many times, there is a need to spot a particular spoken word called as a keyword in the spoken utterance. The system designed for this particular task is called as Key Word Spotting(KWS) system. This is a specific application of Automatic Speech Recognition(ASR). Therefore, most of the approaches used for ASR could also be used for KWS system with a little modification. In this paper, we are using one such ASR approach for KWS system called as Dynamic Time Warping (DTW) technique based on Dynamic Programming(DP) algorithm. Unlike other techniques used for ASR, this does not need training of any models as well as any addition of new keyword to the database do not call for a retraining. The basic approach is keywords are stored as templates in the database and matched against the unknown spoken utterance for similarity. Based on the similarity index, conclusion is drawn that a keyword is found or not.

As with the development and worldwide adoption of ASR technology, additional ASR applications came into picture. Some of the emerging applications that could use KWS system are Audio Indexing, Call Monitoring for National Security, Interactive Voice Response (IVR) system. Basically keyword spotting (KWS) referred to as a problem of searching of keyword template in an unknown speech signal. This task of searching is very important and some of the application which does not require knowledge of whole contents of the unknown speech signal. In such cases we could use KWS system.

In this paper different keyword spotting methods are surveyed. The various approaches used for implementing KWS are, DTW based, Hidden Markov Model (HMM) based and phone or word based. Out of all the above listed methods, one of the easiest strategy for keyword spotting is introduced by Bridle[1] and which is used in [2,3,4] which suggest the use of DTW to search for match between keyword template and test utterance. But the major problem with DTW based approach is its computational complexity and estimation of threshold [5] and poor modelling of word duration [6], which means that we don't know actual starting point and ending point of the word to be spotted and there may be silent part in between. HMM is normally used for ASR, has also been used for keyword spotting. The fundamental idea of this technique, is to build HMM model for both keyword and test utterance. The models other than the keyword is referred as garbage model or filler model and the probability is calculated for each speech region to search if it is closer to the keyword. One more strategy as stated above which involves analysing and searching phone or word based speech recognizer to spot keyword occurrence [5,6]. This strategy is based on speech recognition which involves speech recognizer to spot keyword from predefined vocabulary which gives high error rate [6,7] when the unknown speech is noisy.

Because of these problems associated with the DTW and word-phone based, HMM model is used in most of the previous work of keyword spotting [8]. On the other hand, HMM based keyword spotting suffers from number of problems like mainly collection of large amount of illustrated training data[9]. This annotation alone is time consuming and it requires language expertise. One more problem associated with HMM is regarding

flexibility[10], addition of new keyword which requires retraining. There were many attempts to overcome these problem of HMM-based KWS such as building different types of model[11]. However, it again requires training which can cause a problem for audio indexing of new words, which is the main focus of this paper. These problems related with the HMM, in recent years leads to the use of DTW based KWS.

Because of the recent advancement in the computing power, DTW based system is revisited. In this paper, we have proposed alternative methods related to the sliding window search, in order to reduce the computational complexity. We have extracted Mel Frequency Cepstral Coefficients(MFCC) features from keyword and from unknown utterance. Then these features of keyword and utterance are used by the proposed DTW.

The rest of the paper is structured as follows: Section 2 briefly describes the system design which includes overview of the system, feature extraction method. Section 3 talks about the proposed modified DTW methods using hashing technique. Section 4 describes the experimental setup and the results. Section 5 provides the conclusion and future scope of the system.

II. SYSTEM DESIGN

A. Overview of the System

The block diagram of the keyword spotting system is shown in Fig.1. Feature Extraction Block(FEB) uses MFCC feature extraction method. Feature extraction of keyword as well as unknown utterance is done using FEB. Features

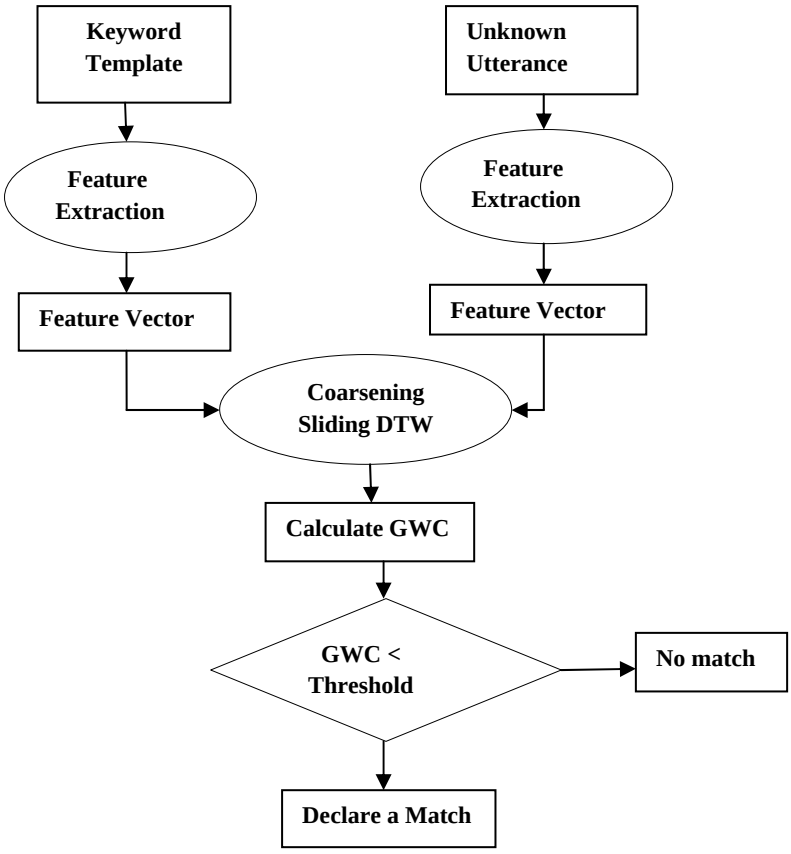


Fig. 1. Proposed Configuration of Keyword Detection System

of keyword and unknown utterance are compared in the modified DTW block. Then the Global Warping Cost(GWC) is computed for each window comparison. If the GWC of the particular window is below the threshold, then it is declared as keyword found.

B. Feature Extraction

Speech signal is applied through a pre-emphasis filter, framed for 25ms, windowed using Hamming widow with an overlap of 15ms. Fourier transform of each frame is applied through a set of Mel Filter banks. Filter bank output is calculated in decibels and applied with a discrete cosine transform to decorrelate the cepstral coefficients. The whole feature extraction method is shown in Fig.2.

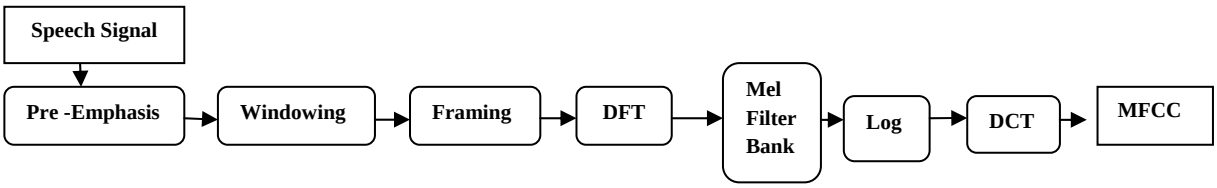


Fig.2. MFCC Feature Extraction

MFCC can be defined as short time power spectrum of speech signal in decibels. So basically MFCC represent the energy in each band of triangular filter. In case of MFCC, Mel scale (m) is used which approximates the human auditory system's response more closely than the linearly space frequency bands. Mel scale warps the frequency and allows better representation similar to human auditory system. Mel scale is roughly linear below 1000Hz and is non-linear (logarithmic) above 1000Hz. So here at the output we get 20 coefficients, but typically lower 13 coefficients are chosen by spectral smoothing. In this paper, we extract 13 static MFCC coefficients, 13 delta coefficients, 13 acceleration coefficients for a speech frame of length 25ms.

III. PROPOSED DTW METHODS

As it is well known that in speech processing, when we record the speech signal, two occurrence of the same word, even if it is uttered by the same person, they are not exactly the same. This is the biggest challenge while designing any ASR system. This could be because of many factors like speaker variation, accent, pronunciation and further may be because of noise while recording the speech signal. The proposed system depends on one hypothesis that the GWC between the keyword and part of the utterance containing the keyword are small compared to other part of the utterance.

A. DTW Algorithm

Assume the keyword feature sequence is represented by (A_1, A_2, $A_3 \ldots A_n$), where n is number of frames in the keyword and unknown utterance is represented by (B_1, B_2, $B_3 \ldots B_m$), where m is the number of frames in the unknown utterance in which m>n. Each A_i, B_i represents frames of keyword and unknown utterance which in turn contains 39 MFCC feature vectors. So the total number of features in keyword, unknown utterance is n*39, m*39 respectively. To compare time series sequence of different lengths, the sequences must be warped in dynamic manner [12,13]. The DTW algorithm will find out warping path between keyword and utterance to be tested. Here we are trying to find out whether there is a keyword present in the long spoken utterance, so the number of frames in the utterance(m) is going to be always bigger than n. The computational complexity of DTW is $O(N^2)$ where N is the maximum length of the two time series.

A generic DTW algorithm is explained below.

Let the keyword template of A=(a_1, a_2, $a_3 \ldots a_n$) and utterance B=(b_1, b_2, $b_3 \ldots b_m$). The absolute distance between the two elements a_i, b_j is d_{ij}. This results in a local distance matrix of length n*m as given by the following equation:

$$dij = |ai - bj|, i = 1,2 \ldots n, j = 1,2 \ldots m. \quad (1)$$

The global distance matrix calculated from the local distance matrix through the following steps.

1. Start with the calculation of *a(1,1) = d(1,1)*
2. Calculate the first row

$$a(i,1) = a(i-1,1) + d(i,1). \quad (2)$$

Calculate the first column

$$a(1,j) = a(1,j) + d(1,j). \quad (3)$$

3. The second row to the last row is calculated by the following steps

$$a(i,2) = min(a(i,1), a(i-1,1), a(i-1,2)) + d(i,2). \quad (4)$$

4. Carry on from left to right and from bottom to top with the rest of the grid

$$a(i,j) = min(a(i,j-1), a(i-1,j-1), a(i-1,j)) + d(i,j). \quad (5)$$

5. Trace back the best path through the grid starting from a(*n*, *m*) and moving towards a(1,1) by following the minimum score path.

Then the GWC is given by

$$GWC = \frac{1}{N}\sum_{i=1}^{P} W_i \quad (6)$$

Where W_i is the cost of the element along the warping path and N = m+n .

B. Complexity Reduction DTW methods

As explained in the earlier section the computational complexity of DTW is $O(N^2)$ where N is the maximum length of the two time series sequence. In general, as given in[2], the local distance computation is done between the keyword of 'n' frames with feature vector per frame as '39' then n*39, the first n*39(1to n frames) of unknown utterance of 'm' frames and the local cost is computed. In the next iteration, the matching is done between n*39 keyword and (2 to n+1)*39 of unknown utterance. The keyword is slide through the unknown utterance frame by frame. Based on this, the local cost of each sliding window is computed. If we had to find out the global cost then the number of iterations is m-n and hence the computational complexity will be $O((m-n)(n*39)^2)$.

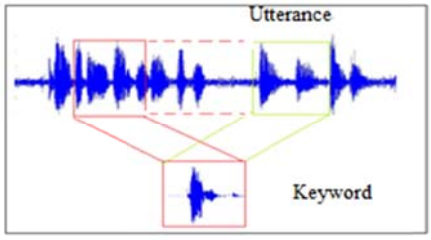


Fig. 3 Sliding Window of DTW

We propose a modified sliding window, whereby instead of sliding the keyword frame by frame, first iteration (matching) is done between n*39 keyword and (1 to n)*39 of unknown utterance. Next iteration is done between n*39 keyword and (n+1 to 2n)*39 of unknown utterance. This is shown in Fig. 3. Hence the number of iteration is reduced to ceil (m/n) and hence the computational complexity is reduced to $O((ceil(m/n))(n*39)^2)$.

We propose a technique similar to hashing technique used in search algorithms of data structures, in which the sequence of the keyword which is represented by (A_1, A_2, A_3…A_n), where A_1 is sum(a_i) where i varies from1 to 39. In generic methods A_i in turn represents sequence of feature vectors per frame. Here, we represent A_i as a summation of all the feature vectors per frame. In this way the length of the sequence as well as the search space is reduced by 39. So the length of the keyword sequence is n instead of n*39. Similarly the unknown spoken utterance by (B_1, B_2, B_3…B_m), where B_1 is sum(b_i) and i varies from1 to 39. So the length of the keyword sequence is m instead of m*39. DTW algorithm is performed for m-n iterations, but the complexity is highly reduced to $O((ceil(m/n))(n)^2)$.

To find out the match, threshold estimation is needed. Once the GWC is found out and if the GWC of that particular window is below the threshold , then the keyword is spotted in that place. Here we employ the threshold as

$$T = Mean - (C * STD) \quad (7)$$

Complexities of various experiments conducted in section 4 is given in Table I.

TABLE I
Complexity Comparison of Proposed Methods

Proposed Methods	**Sliding Window**	**Coarsening of Feature Vectors**	**Window Length**	**DTW Matrix Size**	**Number of GWC Calculations**	**Complexity**	**Time Taken**
Experiment1	yes	yes	one keyword	NxN	M/N	(M/N) $O(N^2)$	< 1 sec
Experiment2	yes	No	one keyword	39Nx39N	M/N	(M/N) $O((39*N)^2)$	5 mints
Experiment3	yes	No	one keyword	39x39	39(M/N)	M $O(39^2)$	1-2 sec
Where N is the number of frames in the Keyword, M is the number of frames in the spoken utterance, number of feature vectors per frame is 39.							

IV. EXPERIMENTAL SETUP AND RESULTS

A. Experimental Setup

In this work, we have experimented with *TI Digit*[16] - connected digit data to spot the individual digit as keyword in the connected digit which was treated as the spoken utterance. The Corpus contains isolated digit files , connected digit files for each speaker. *1a.wav* keyword template to be searched in the *5o217a.wav*. First the MFCC feature vectors which includes static, delta and accelerated coefficients are extracted from the keyword and from the spoken utterance using HTK Tool kit[14]. They are aligned as one dimensional vector sequence as frame by frame. Using the proposed adapted sliding window method along with the reduced sequence length using hashing technique, GWC was computed and thereby found out whether the keyword was present or not. The experiment was carried out for both speaker independent, speaker dependent category. Speech files used are from Connected TIDIGIT data test\man\ar and test\man\ah. Four spoken utterances such as 3o33951, 5o217, 24z982z and 27o6571 are selected from the test corpus of TI Connected Digit data randomly. For these spoken utterances, keyword digits from 1 to 9 are searched whether they are being uttered or not using the proposed methods.

B. Results

1)Experiment 1

Using the coarsening sequence technique and adapted sliding window DTW method, the GWC was calculated. Using the equation(4) and assuming the value of C is 1, keyword detection is performed for the connected digit utterance.

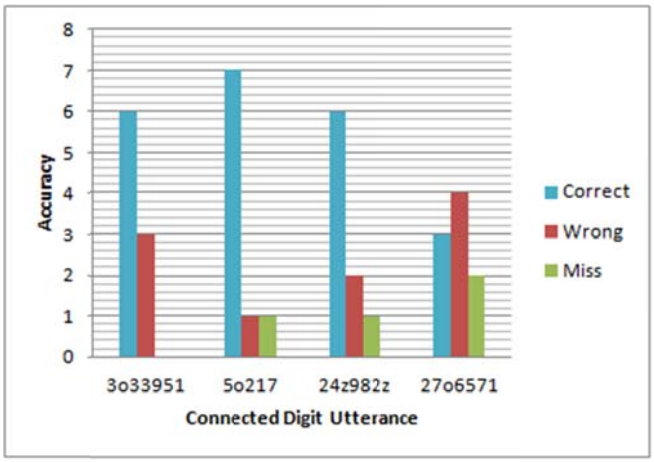


Fig.4. KWS Results of Experiment 1 for Speaker Dependent.

For the spoken utterance 5o217, for the keyword digits 1,2 the GWC is lesser than the threshold value, so it is treated as keyword being found which is "correct". For the keyword digits 3, 4 the GWC is greater than the threshold value and so it is being treated as keyword not being found which is not spoken in the utterance as per the example treated as "correct". For the keyword digit 5, the GWC is greater than the threshold value that means keyword not spoken but as per the example the keyword being spoken in the utterance, so it is treated as "miss", means missed to detect. For the keyword digit 8, the GWC is lesser than the threshold value that means keyword being spoken but as per the example the keyword not spoken in the utterance, so it is treated as "wrong", means wrong detection. The spoken utterance '5o217' uttered as 'five oh two one seven' checked with the 1-9 digit keyword templates resulted in 7 correct , 1 wrong, 1 miss. Similarly, this experiment is carried out for the other spoken utterance and the results are shown in Fig. 4, 5. Fig. 4 shows the results of speaker dependent search. Fig. 5 shows the results of speaker independent search.

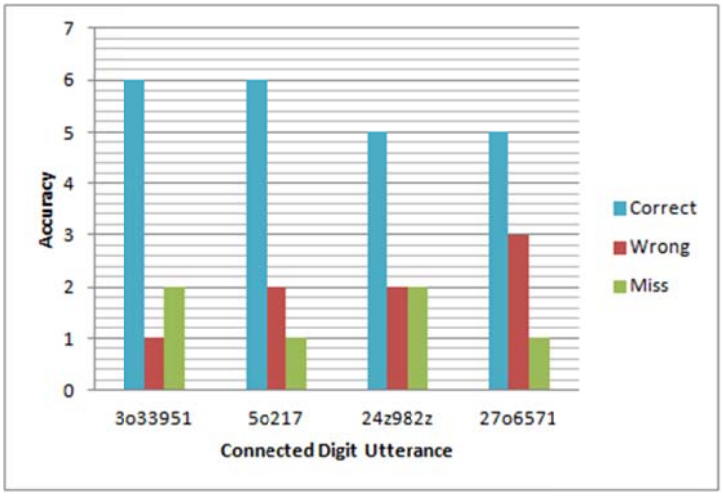


Fig.5. KWS Results of Experiment 1 for Speaker Independent.

2)Experiment 2

In this experiment, the feature vectors from the utterance and keyword are considered without the coarsening technique. Here the DTW matrix size is 39 times bigger than the size of the matrix that we have considered for experiment 1. So the time taken to compute the GWC was considerably larger around 5 minutes. Using the equation 4 and the value of 'c' as 1, the threshold value is computed and thereby detected whether there is a match or not. The accuracy of the different utterance results are shown in Fig. 6.

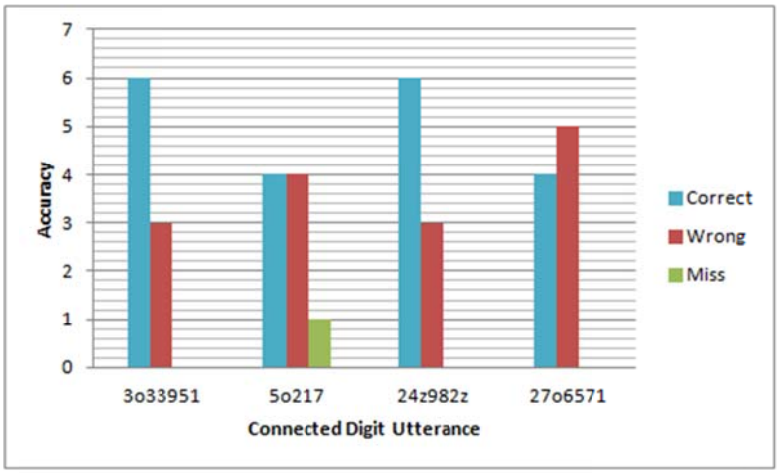
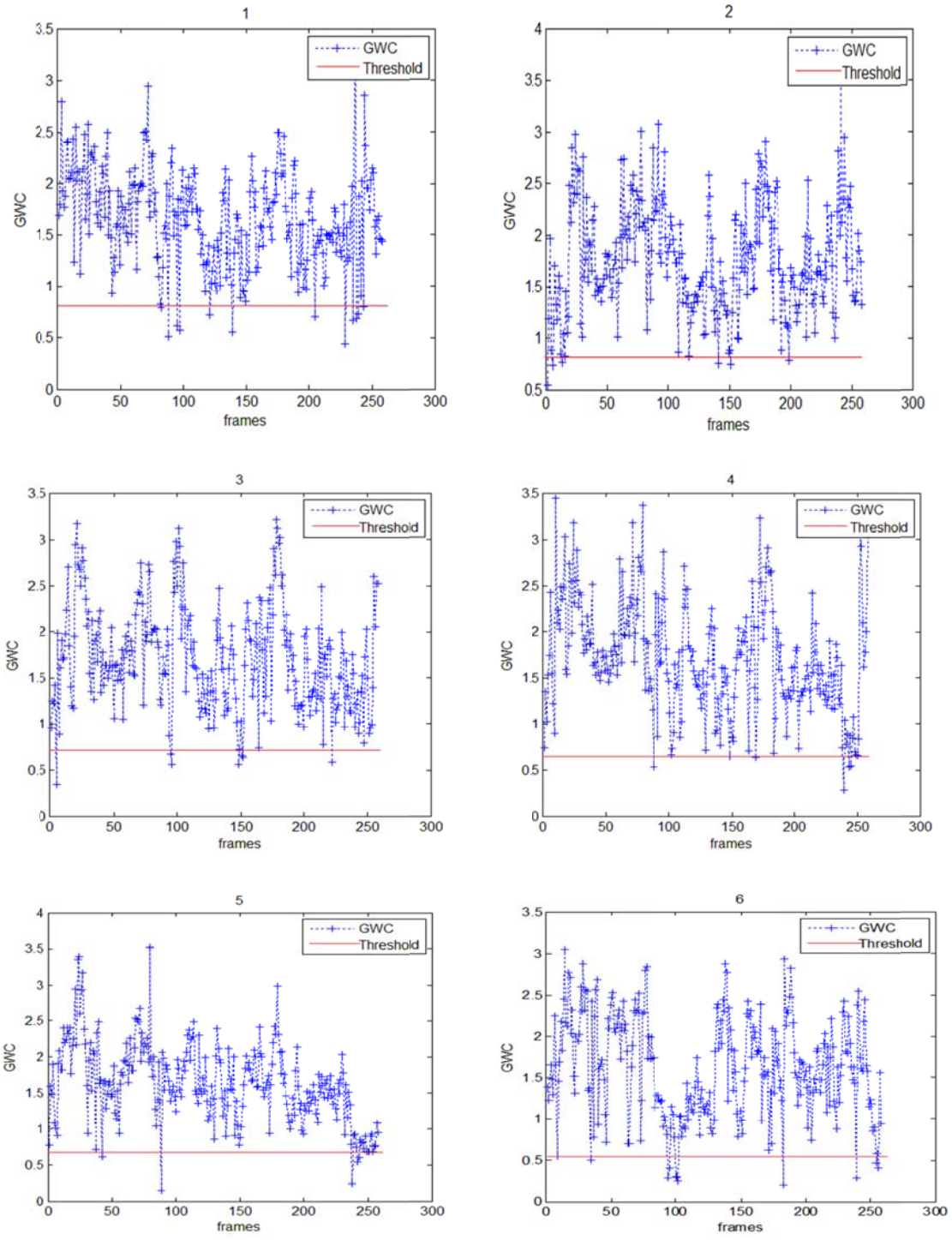


Fig.6. KWS Results of Experiment 2 for Speaker Dependent

3)Experiment 3

Consider the keyword contains 'n' number of frames, Utterance contains 'm' number of frames. For one keyword length of the utterance, ie,{1 to n}of 'm' frames, frame by frame DTW algorithm is performed and the GWC is computed. Next comparison is done with {n+1 to 2n} frames of utterance against keyword. Thus GWC sequence is computed. Here threshold value is considered as mentioned in equation (4) with the value of 'c' as 2. If 2 or more GWC is below the threshold value is found then it is treated as keyword is found in the utterance. Fig. 7 shows GWC plot for each keyword digit search in the spoken utterance 2706571a. Fig. 8 shows the accuracy of this experiment for the same spoken utterance.


1
2
3
4
5
6
GWC
Threshold
frames

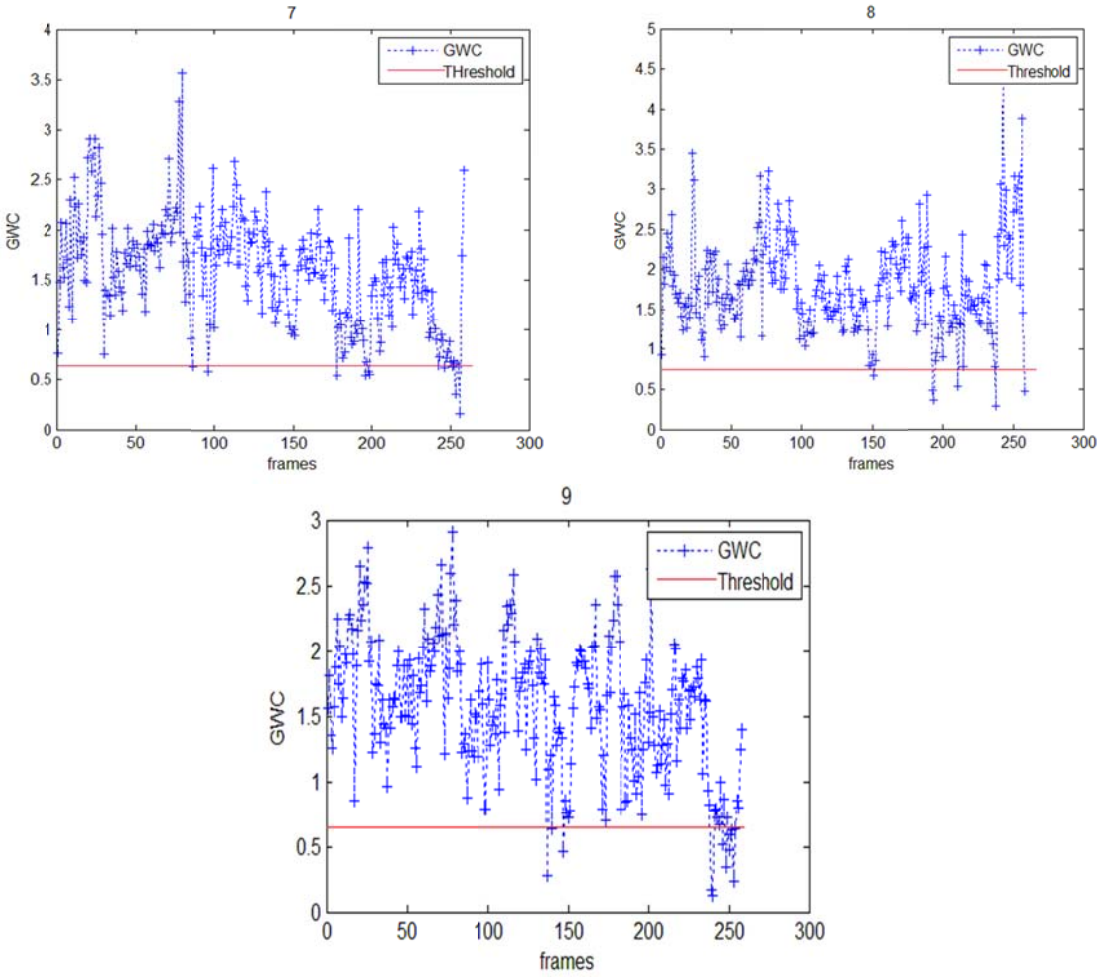


Fig. 7. Plot of GWC for each keyword digit search for the spoken utterance 2706571a

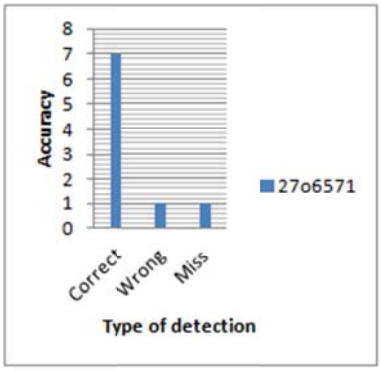


Fig. 8. KWS Results of Experiment 3 for the Spoken Utterance 2706571a

V. CONCLUSION AND FUTURE SCOPE

The proposed techniques along with the DTW to detect the keyword in the unknown spoken utterance were experimented. These methods indentified to spot keyword without the need for the training of filler and garbage model as in HMM based technique. Because of the evolution in the computing power, the disadvantage of computational complexity of DTW also vanishes. In addition to this, this paper proposed various reduced computational complexity methods of DTW. The projected experiment1 gives better performance as compared to other experiments and also shows efficient. Further, this process can be implemented for twice-length and half-length of the keyword while comparing with unknown utterance using the proposed methods. The efficacy of the methods could be tested more with the different feature extraction techniques of speech.

REFERENCES

[1] J. Bridle, "An efficient elastic-template method for detecting given words in running speech," in Proc. Brit.Acoust. Soc. Meeting, 1973, pp. 1-4.

[2] M. S. Barakat, C. H. Ritz, D. A. Striling "Keyword Spotting Based on Analysis of Template Matching Distances", 978-1-4577-1180-0/11/$26.00 2011 IEEE.

[3] A. Higgins and R. Wohlford, "Keyword recognition using template concatenation," Proc. ICASSP '85. 1985, pp.

[4] R. W. Bossemeyer, J. G. Wilpon, C. H. Lee, and L. R.Rabiner, "Automatic speech recognition of small vocabularies within the context of unconstrained input,"The Journal of the Acoustical Society of America, vol. 84,p. S212, 1988.

[5] Y. Peng and F. Seide, "Fast Two-Stage Vocabulary-Independent Search In Spontaneous Speech," Proc.ICASSP'05, 2005, pp. 481-484.

[6] F. Seide, Y. Peng, M. Chengyuan, and E. Chang,"Vocabulary independent search in spontaneous speech,"Proc. ICASSP '04, 2004, pp. I-253-6 vol.1.

[7] R. Ordelman, F. de Jong, and M. Larson, "Enhanced Multimedia Content Access and Exploitation Using Semantic Speech Retrieval," in Semantic Computing, 2009. ICSC '09. IEEE International Conference on, 2009, pp.521-528.

[8] J. Keshet, D. Grangier, and S. Bengio, "Discriminative keyword spotting," Speech Communication, vol. number 51, pp. 317-329, 2009.

[9] Z. Yaodong and J. R. Glass, "Unsupervised spoken keyword spotting via segmental DTW on the Gaussian posteriorgrams," in Automatic Speech Recognition and Understanding, 2009. ASRU 2009. IEEE Workshop on, 2009, pp. 398-403.

[10] M. Wollmer, B. Schuller, F. Eyben, and G. Rigoll, "Robust vocabulary independent keyword spotting with graphical models," in the Automatic Speech Recognition &Understanding, 2009. ASRU 2009. IEEE Workshop on, 2009, pp. 349-353

[11] Ryuichi Oka, "Spotting Method for Classification of Real World Data" The Computer Journal, Vol.41, No.8, 1998

[12] Titus Felix FURTUNĂ, "Dynamic Programming Algorithms in Speech Recognition" Revista Informatica Economica nr. 2 (46) / 2008

[13] MarutiLimkar, RamaRao & VidyaSagvekar "Isolated Digit Recognition Using MFCC AND DTW" International Journal on Advanced Electrical and Electronics Engineering, (IJAEEE), ISSN 2278-8948, Volume-1, Issue-1, 2012

[14] http://htk.eng.cam.ac.uk

[15] S. Young, J. Jansen, J. Odell, D. Ollason, and P. Woodland, The HTK Book, Cambridge Univ., 1996

[16] http://catalog.ldc.upenn.edu/LDC93S10