

Optimizing Algorithms for Phrase Recognition

J S Mirza , Muhammad Umair

Department of Computer Science

COMSATS Institute of Information Technology, Lahore, Pakistan

jsmirza@ciitlahore.edu.pk, muhammadumair@ciitlahore.edu.pk

Abstract

The paper presents a simple algorithm to cull short phrases of interest spoken in a long recorded speech. Listening to a long and boring recording of a suspect who is being wiretapped may irk intelligence investigator if it does not contain any speech segment of interest. It would be much useful if first a preliminary test is run to see if the recorded speech does contain phrases of interest which could help convict a suspect. If so the entire recording can then be reexamined. Vowels spoken in an utterance can be reasonably identified by locating the vowels' first two formants i.e., F1 and F2 onto vowel loops similar to the ones drawn by Peterson and Barney for English vowels' sounds [1]. Using Pratt or a similar type of software can determine the formants of vowels. Two algorithms were designed to map the vowels' formants spoken in an utterance onto the vowel loops and identify which vowel sounds they represented [2]. The algorithm-1 used the calculated value of

F1 and determined in which vowel-loop F2 lied, and algorithm-2 used the calculated value of F2 and determined in which vowel-loop F1 lied. In some cases the algorithm-1 did a better job than algorithm-2 in terms of computational time while in other cases the reverse was true. This paper essentially extends the idea of mapping (F1,F2) onto the vowel loops by another algorithm, called algorithm-3, which is faster than the previous two. The algorithm-3 requires conversion of vowel loops in a particular format to act as our data bank; let us call it conversion table because it will convert any point F1,F2 that is covered by the vowel loops into a corresponding vowel symbol. The calculated values of F1 and F2 of vowels are discretized in suitable steps, and a corresponding vowel symbol is determined from the conversion table. The vowel symbol at the intersection of F1 and F2 is selected.

1. Introduction

Speech recognition is an uphill task and despite rigorous progress in available techniques the success rate is insufficient in special circumstances to have an acceptable recognition result. Further, in most of the speech recognition system it is imperative that speaker first train himself on the recognition system before his utterances can be reasonably recognized. An untrained system most likely will go haywire. In other words if someone is wiretapping somebody's phone line and that line has access to a number of users, as might be the case, or if the same line is used even by one speaker, and the recognition system has not been trained on his/her voice then the speech

recognition will prove to be difficult. This is because normal recognizer necessarily requires training of speakers on the system. The training helps recognizer system get the data of formants of vowels and typical characteristics of consonants spoken by the speaker. There are lots of parameter one has to consider for efficient speech and speaker recognition [3].

In this paper a simple technique is presented to see if the recorded speech does contain phrases of interest. If so, the entire recording can then be reexamined. Once a given short phrase has been recognized a cue is received to inspect the entire speech segment for finding what one is looking for. This can become a useful tool for a wiretap per who can ignore listening to long speech

segments, which may happen to be worthless and sniff some important phrases, which can indicate that something important may be present in the recorded speech.

The scheme requires first storing the spoken desired phrase(s), extracting the formants of their vowels and save them in the same sequence as is found in the phrase. The extraction was done by the use of Pratt. The location and duration of vowels' in the recording was based on pitch information. The time span of the pitch gives the requisite location and duration of the vowels. This constituted our databank. The recording in which the desired phrase is looked for is analyzed for its vowels and matching with the databank is implemented. Analysis of the recording is confined only to sequential recognition of all of its vowels. The sequence of vowels is very important. Vowels recognition is done by extraction of first two formants which are F1 and F2. and mapping them on vowel loops

Subsequent to formation of databank, the scheme proceeds to first extract the first two formants, F1 and F2 of all the vowels contained in the recording, which is to be analyzed. The consonants are ignored, because of their difficulty in analysis and to simplify the scheme. Any suitable software like Pratt can be used for this purpose. Irrespective of whosoever speaks the F1 and F2 of vowels normally fall in their respective loops as is evident from English vowel loops (Peterson and Barney [1]. There is a minor overlapping between loops of certain vowels, and therefore formulation of the vowel loops can be somewhat problematic. For instance out of 10 vowel loops in English the following loops overlap by a maximum of about 20%; (E and OO), (A and OW), (U and ER); the vowels in parenthesis are presented in their typewritten symbols for convenience and not in IPA symbols. It must be pointed out similar overlapping takes place in Urdu Vowels also. The overlapping can be handled by another scheme. For convenience and due to lack of viable data regarding Urdu vowel loops it is assumed that English vowel-loops

and their counterpart vowel-loops in Urdu are exactly the same; even though our initial investigation does indicate minor differences. These differences, however, are minor and therefore negligible for the vowel set we are interested in for our testing. An experiment is currently underway at our facility to determine vowel loops of Urdu. The English vowel loops that we have employed are those of Peterson and Barney [1].

2. Databank and Algorithm

The algorithm uses the databank as mentioned before which contains the information obtained from the vowel loops in a special format. English vowel loops [1] are shown in Figure 1 to provide explanation. The vowel loops are covered by a close-knitted grid, not shown in the diagram to avoid mess, The horizontal and vertical lines of the grid are discredited by a suitable step size. The step size we chose was 50Hz along F1 and F2 axes. The algorithm-3 could adjust its data for any step size, which is a multiple of 50 Hz if it is desired to change the step size to expedite recognition process. The calculated values of F1 and F2 of vowels are translated into corresponding discredited values. The intersection of F1 and F2 lines will select the vowel symbol, which lies there. The algorithm-3 is presented in **Figure 2**.

Algorithm-3(display)

Data:

Char data[1400][4000]
Char symbol[10]

Display ()

- Initialize the "symbol" with Possible characters
- For(I =0; i<10;i++)
 - DrawShape(data,symbol [i])
- Stop

Algorithm-3(read data)

Data:
char data [1400][4000]
int intervalf1 =1, intervalf2 = 1

ReadData ()

- Initialize the “data” with “space”
- Message “Option to change default interval”
- Read “Option”
- If “Option is “Yes” then
 - Read “intervalf1”
 - Read “intervalf2”
- For(i=0;i<=1400;i+=intervalf1)
 - For(j=0;j<=4000;j+=interval f2)
 - Read data[i][j]
- Stop

Algorithm-3(find symbol)

Data:
char data[1400][4000]
int f1, f2;

FindSymbol()

- Read f1, f2
- Symbol = data [f1][f2]
- Display “Symbol”
- Stop

Algorithm-3(draw shape)

Draw Shape (char data [][], char ch)

- For(i=0;i<=1400;i+=intervalf1)
 - For(j=0;j<=4000;j+=intervalf2)
 - If (ch == data[i][j])
 - Display “ch” at reference (i , j)
- Stop

If software calculates (F1 and F2) after

short intervals then a great many such points will result. We could either select the vowel symbol for each of the points which could be a bit cumbersome because so many points will be at hand for just one vowel or design an algorithm which would use selective points to decide which vowel they correspond to.

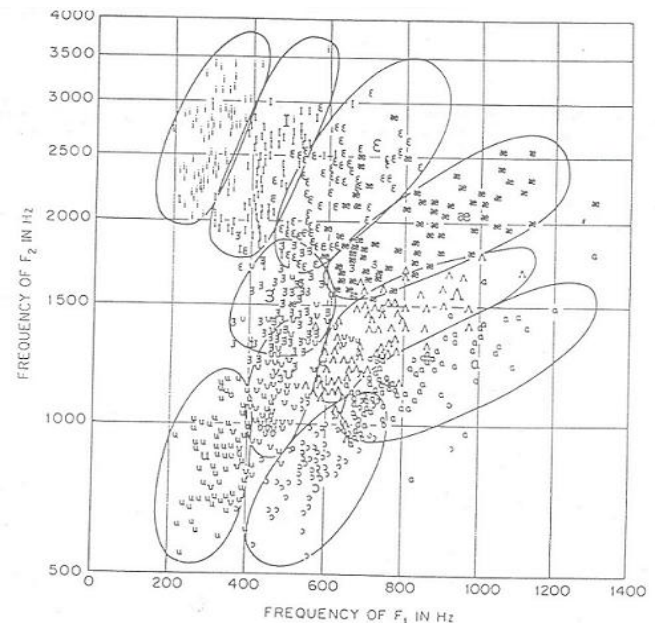


Figure 1: Plot of second formant frequency versus first formant frequency for vowels by a wide range of speakers. (After Peterson and Barney)

Further agility in decision-making could be achieved if instead of the Cartesian coordinates of F1 and F2, polar coordinates are used. The number of comparisons to determine could further be reduced thus enhancing the agility. Each vowel loop of Peterson and Barney can be described by two angles which depict the range within which the vowel lies. Another way to achieve higher speed in recognition process may be to adjust, if we are using algorithm-1, the F1 steps such that at the junction where two vowel-loops meet the F1 steps decrease automatically while for F1 in the middle of the loops, its incremental step increases automatically. For algorithm-2 the adjustment will be for F2. This scheme will require some kind of feedback [4].

3 Experiment and results

The algorithm-3 and the conversion table was tested on a phrase “Usama Bin Laden”

with no context surrounding it. The average success rate of correct determination was 82 % which happens to be nearly the same as for Algorithm-1 and algorithm-2 [2]. The reason is that databank used in both the experiments are the same; the only difference being in the format of the databank. Same data which was used by algorithm-1 and algorithm-2 was used by the algorithm proposed in this paper. Five male speakers aged between 27 and 48 uttered five times each the above phrase “Usama Bin Laden” into a microphone connected to a computer running Pratt software. Formants F1 and F2 were derived by Pratt software. The values available at the centre points of vowels were taken to stay clear of the effects of transitory regions surrounding each vowel.

This algorithm-3 picked only vowels and no consonants, uttered in a short phrase, and represented them in a sequence. The vowels and their sequence were used to determine if the desired phrase had been spoken. Recognition of consonants as well, undoubtedly, can enhance recognition score but it is a cumbersome task, though not impossible.

4. Discussion

Many questions need to be answered. A short phrase, one is looking for, in a recording evidently will contain just a few vowels only, say four, five etc. There is enough probability that another phrase totally different from the one, which is being searched may be present in the speech thus causing confusion. Also there is a possibility that a given vowel voice could be construed wrongly as a different vowel—this may be true in those cases where the vowel formants F1 and F2 are close to each other as in the case of (E and OO), (A and OW), (U and ER) for instance. Thus a phrase-of-interest may be present but due to wrong identification of some of its vowel could not

be searched. A good solution in this case might be to have all the probable versions of a phrase ready in the computer for matching. The probable versions can be found by investigating which vowel sound could be picked up as more than one vowels. Chances are that some vowels sounds as (E and OO), (A and OW), (U and ER) could be misinterpreted as two vowels. Even some vowels sounds could be misinterpreted as one of the four vowel especially those whose F1 F2 are nearer to each other as for instance: ER can be misinterpreted by the analysis software as U,E,AE and I. The brain is much more adept at analyzing the sound by looking at the context of the phrase and therefore the chances of error are close to nil while analyzing software works only on the phrase and not on the context thus enhancing the misinterpretation level.

4 Conclusion

A scheme to determine whether a desired phrase has been spoken in a short utterance, a sentence or a longer text can be determined by using a speech analysis software and conversion table derived from Peterson and Barney vowel loops. Only vowels were used for recognition. The recognition score though not high, presently, can be enhanced by including consonants in our process. It may be noted that in some vowel-loops (figure.1) more than one vowel sounds creep in; also some loops overlap. This drags score of recognition lower. Some vowels are straightforward and do not add to confusion e.g., “IY”, “I”, “AE”. If a phrase, required to be recognized, contains straightforward vowels, recognition score will be higher than those cases where vowel-loops contain more than one vowels. We may, however, consider all the possible sequences, though some of them will be faulty. If the cue from the result is sound enough the entire recording can be examined, else not.

References:

- [1] Peterson, G.E., and Barney, H.L., “Control Methods Used in Study of the

Vowels”, J. Acoust. Soc. Am., Vol. 24,
No.2, pp175-184, March 1952

- [2] Mirza J.S and Hayat S. A., Wire-tapped Intelligence; Machine Recognition of Specific Phrases to Nab A Suspect. *Accepted for publications in the 2005 International Conference on Software Engineering Research and Practice (SERP'05 JUNE 27-30, 2005, Las Vegas, USA)*
- [3] Minh N.Do, An Automatic Speaker Recognition System, *Digital Signal Processing Mini-Project, Audio Visual Communications Laboratory, Swiss Federal Institute of Technology, Lausanne, Switzerland. 2003*
- [4] Dag Stranneby, Digital Signal Processing – DSP and applications, *Newnes publishers, 2001*