Effects of emphasizing transitional or stationary parts of the speech signal in a discrete utterance recognition system

Elenius, K. O. E. and Blomberg, B.

journal: STL-QPSR
volume: 22
number: 4
year: 1981
pages: 039-048

http://www.speech.kth.se/qpsr
III. SPEECH ANALYSIS

A. EFFECTS OF EMPHASIZING TRANSITIONAL OR STATIONARY PARTS OF THE SPEECH SIGNAL IN A DISCRETE UTTERANCE RECOGNITION SYSTEM

Kjell Elenius & Mats Blomberg

Abstract

A pattern matching word recognition system has been modified in order to emphasize the transient parts of speech in the similarity measure. The technique is to weight the word distances with a normalized spectral change function. A small positive effect is measured. Emphasizing the stationary parts is shown to substantially decrease the performance. Adding the time derivative of the speech parameters to the word patterns improves performance significantly. This is probably a consequence of an improvement in the description of the transient segments.

Introduction

The main strategy for isolated word recognition is a pattern matching scheme, combined with dynamic programming for time alignment. The parametric input is either LPC-coefficients or filter bank amplitudes.

Davis and Mermelstein (1) used an MFCC-analysis method (mel frequency cepstrum coefficients) based on the frequency resolution characteristics of the human ear. They concluded that the results using MFCC "indicated superior performance" compared to a 10th order LPC representation using the Itakura metric (2). There was little degradation in their recognition results when using 6 MFCC-coefficients instead of 10.

The pattern matching method has proved to be very useful for isolated word recognition. It is conceptually straightforward and easy to implement. However, it has the inherent feature of assigning equal weight to every time sample of an utterance. But speech, on the other hand, is known to be composed of stationary parts and transitions, and much of the perceptual information of the speech signal is concentrated around the transitional parts (3,4,5). It is possible that in more difficult tasks, such as speaker-independent recognition, giving emphasis to transitional parts might improve system performance.

*This paper will be presented at the 1982 IEEE International Conference on ACOUSTICS, SPEECH and SIGNAL PROCESSING in Paris (ICASSP82).
Some research has been reported using modified traditional pattern matching schemes. Tappert and Das (6) omitted highly correlated samples, achieving a 50% storage reduction without losing recognition accuracy. Das (7) introduced a "variation threshold", and wherever the absolute norm difference between successive samples exceeded the threshold, one or more samples were inserted by interpolation to keep the variation between successive samples below the threshold. Increasing the number of samples by 45% resulted in a slight improvement of recognition performance; also compare work by Mlouka and Lienard (18).

To improve the performance of the pattern matching approach for acoustically similar words, Rabiner and Wilpon (8) suggested a two-pass strategy. First, the unknown utterance is designated to a class of acoustically similar words using ordinary pattern matching. In the second pass, the word is matched within the class using a statistically based weighting function, which emphasizes the acoustically dissimilar parts. The method requires considerable processing but resulted in a significantly improved recognition accuracy.

Another method of focusing the interest to the transitional parts of the speech wave is reported by Kuhn et al (9). They used a technique where the total spectral change over an utterance is equally distributed among the samples, i.e., the spectral change between all samples is the same. Every pattern is nonlinearly warped onto a uniform format. They report dramatic reductions in computer time (factor 9–22) and memory requirements (factor 3–4) without degrading the performance compared to conventional pattern matching. The same technique was used by Scope Electronics Inc. (10) and Yu-Tie Chen (11).

Blomberg and Elenius (12) have investigated the contribution of dynamic features to a discrete word recognition system based on phoneme segmentation. A phoneme and diphone library was collected using 6 male speakers and a vocabulary of 41 words. The steady state portion of each phoneme was represented by midpoint values of broad band energy levels and center of gravity measures for formant frequency approximation. The transient part between two phonemes, the diphone, was represented by the time difference of these parameters in an interval across the segment boundary. The position and duration of the interval relative to the boundary were varied to maximize the discrimination between the diphones. The discrimination function used was F-ratio (13). The
results are shown in Fig. III-A-1 for three parameters. A symmetric position of the interval and a duration of 40 ms for low frequency energies and 100 ms for higher frequency energies maximized the F-ratio. Adding the diphones, i.e., the dynamic features to the phoneme library substantially improved the recognition rate.

**Dynamic programming method used**

The method we have used for calculation of distances between word patterns is the "normalize and warp" procedure as described by Myers et al (14). Every input word is linearly normalized to a fixed length of N samples. Then we apply a dynamic programming algorithm as proposed by Sakoe and Chiba (15). We will review it briefly using their notation. Two speech patterns A and B are represented by a sequence of feature vectors.

\[ A = a_1, a_2, \ldots, a_N \]

\[ B = b_1, b_2, \ldots, b_N \]
The distance $d(i,j)$ between feature vectors $a_i$ and $b_i$ is:

$$d(i,j) = \sum_{p=1}^{P} |a_{i,p} - b_{j,p}|$$

We have used a symmetric DP-algorithm.

$$g(i,j) = \min \left\{ g(i-1, j-2) + 2d(i, j-1) + d(i, j),
\begin{align*}
g(i-2, j-1) + 2d(i-1, j) + d(i, j),
g(i-1, j-1) + 2d(i, j),
g(i-2, j-1) + 2d(i-1, j) + d(i, j) \end{align*} \right\}$$

Our boundary conditions force the endpoints of A and B to be mapped on each other.

An adjustment window $r$,

$$|i-j| \leq r,$$

defines a maximum deviation from the line $i=j$.

The total distance between patterns A and B will be

$$D(A,B) = g(N,N).$$

How to emphasize the transitions

The aim of the experiments reported here is to try some new ways of focusing the interest to the transitional parts of the speech. We want a method that is simple to realize in an ordinary pattern matching framework. To do this we first introduce a time difference function for each speech parameter $p$ at sample point $s$:

$$df_{s,p} = a_{s+d,p} - a_{s-e,p}$$

The difference is thus measured over an interval that is $d+e$ samples long.
We also introduce a total difference function:

\[ DF_s = \sum_{p=1}^{P} |a_{s+d,p} - a_{s-e,p}| \]

This difference function has been used as a measure of the transitional segments, so that local maxima in this function should correspond to transitional speech segments.

The total difference function is normalized,

\[ W_s = \frac{DF_{s-f}}{\sum_{s=1}^{N} DF_{s-f}} \quad \text{to give} \quad \sum_{s=1}^{N} W_s = 1 \]

By varying \( f \), we can translate the weight \( W_s \) \( f \) samples forward or backwards compared to \( DF_s \) and put the emphasis on the samples following or preceding the transition. \( W_s \) is then used for weighting the distances \( d(i,j) \) in the DP-algorithm. Since there is one weighting function associated with each of the patterns A and B, we use

\[ W_{A,i} W_{B,j} \]
\[ d_w = \frac{W_{A,i} W_{B,j}}{2} * d(i,j) \]

as the weighted local distance between points \( i \) and \( j \) in patterns A and B.

The "inverse" of \( W_s \) is defined as

\[ W_s' = \max \left\{ \frac{2}{N-W_s}, 0 \right\} \]

To avoid negative distances, \( W_s' \) is always \( \geq 0 \).
Experiment

The speech is input through a 1/3-octave filter bank (frequency range 20-10000 Hz), with a sampling rate of 50 Hz. The input filter section contains the logarithmic peak amplitude of each filter during the 20 ms sampling interval. The filter spacing is converted to a critical band scale (Bark scale), which is modeled according to the frequency characteristics of the human ear (16,17). The critical band amplitudes are converted to critical band cepstral coefficients (CBCC).

\[ \text{CBCC}_j = \sum_{k=1}^{19} x_k \cos(j(k-1/2) \pi/19) \quad j=1,2,...,P \]

where P is the number of cepstral coefficients and \( x_k \), \( k=1,...,19 \) represents the log-energy output of the k-th critical band. Since the 0-th order coefficient is discarded, the CBCC’s should be insensitive to differences in signal amplitude. However, the 40 dB dynamic range of the filter bank will distort the spectra of low energy signals, making the CBCC’s somewhat sensitive to signal amplitude.

The vocabulary consists of 30 Swedish names, the first name plus the family name, uttered in connection. The references are built as mean values of many speakers. The first input utterance of each reference is used for time alignment of the second, and a mean is interpolated. The third utterance is warped to this mean giving a new reference, and so on. A total of 21 repetitions of each name is used to calculate the references from 10 speakers. The recognition is based on 16 repetitions of the words from 8 speakers, two of which were not in the reference group. All words are stored in the computer after a linear time normalization to 32 samples. The length of the input words are 1.3±0.3 s. This means that the length of a sample interval is 42±10 ms.

We have used the following difference functions:

- \( \text{DF}_1 \) = \( \text{DF}_5 \), \( d=1 \), \( e=0 \), i.e., dif. over 1 sample int.
- \( \text{DF}_2 \) = \( \text{DF}_5 \), \( d=1 \), \( e=1 \), i.e., dif. over 2 sample int.
- \( \text{DF}_3 \) = \( \text{DF}_5 \), \( d=2 \), \( e=2 \), i.e., dif. over 4 sample int.
The weighting functions $W_s$ have been derived from DF1 and DF2 and the weight has been translated -2, -1, 0, 1, 2 sample intervals.

To emphasize stationary parts a weight $W_s$ has been derived from DF2 without any translation.

The functions $df_{s,p}$ have been used as recognition parameters. The notation $df_1$, $df_2$ and $df_4$ means the difference over 1, 2 or 4 samples, cf. DF1, DF2, DF4.

The adjustment window, $r$, has been varied from 2 up to 6 samples.

Results

![Graph](image)

**Fig. III-A-2.** Effects on recognition rate using 6 CBCC's and no weight, $W_s$ (from DF1, DF2) and $W'_s$ (from DF2).

The effects of weighting the distances by functions derived from DF1 and DF2 are shown in Fig. III-A-2. The weights have not been translated. A slight improvement in recognition accuracy compared to ordinary distance calculation is achieved by using a weighting function. Using a weight that emphasizes stationary parts results in substantially lower performance.
In Fig. III-A-3, the time shift of the weight relative to the difference functions (DF1, DF2) is varied. It seems best to put the emphasis right on the transitional parts and it is also seen that placing it before the transition is better than placing it after.

In ref. (12) it is shown that the use of dynamic features improves the performance of a word recognition system. Thus, the difference functions \( d_f \) themselves should be tested as recognition parameters. Using 4 CBCC's resulted in 91.7% correct recognition with \( r = 4 \). Using only the df1's of these CBCC's resulted in 90.6%. Trying to weight the difference parameters resulted in little improvement. Adding the difference functions of the 4 CBCC's resulted in significant improvement, as can be seen in Fig. III-A-4. Measuring the difference over a window of 2 samples (~85 ms) gives a somewhat better performance than a window of 40 or 170 ms. This is in agreement with (12). The reason for using only 4 CBCC's in this experiment was memory limitations of the computer.
From Table III-A-I it can be seen that adding difference functions to 4 or 5 original parameters increases recognition accuracy about as much as increasing the number of CBCC's by 1.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Recognition accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>df1 of 4 CBCC</td>
<td>90.6</td>
</tr>
<tr>
<td>4 CBCC</td>
<td>91.7</td>
</tr>
<tr>
<td>4 CBCC + df2 of them</td>
<td>95.0</td>
</tr>
<tr>
<td>5 CBCC</td>
<td>95.6</td>
</tr>
<tr>
<td>5 CBCC + df1 of first 4</td>
<td>96.9</td>
</tr>
<tr>
<td>6 CBCC</td>
<td>96.5</td>
</tr>
</tbody>
</table>

Table III-A-I. Recognition results using different combinations of CBCC's and their difference functions. Adjustment window, \( r = 4 \).

Conclusion

The use of a weighting function to emphasize the transitional parts of the speech wave has been shown to give a little increase in recognition performance. The accuracy of about 97% without any weighting makes it hard to get any significant improvements. On the other hand, emphasizing the stationary parts lowers the performance significantly. This indicates that essential information for discriminating words is to be found around the transitions. It may be that the technique used for extracting this information is too simple.

Adding the time differences of the CBCC parameters improves recognition with greater significance. The explanation might be that including the dynamic features of the speech adds information about the transitions that is not explicit in the ordinary parameters. The effect may be more pronounced in multi-speaker speech recognition. Still, the use of only time differences gives lower performance than only CBCC's, indicating their use as a complement to the CBCC's.

Adding the summed difference parameter (DF1 or DF2) to the original CBCC's does not improve the performance. This measure evidently is too
crude to capture the information of the transitional parts of the speech wave. This may also explain why the use of it as a measure of transitions showed no greater effect on the performance in the weighting experiments.

References


