Comb Filter Decomposition for Robust ASR

Lech Szymanski, Martin Bouchard

School of Information Technology and Engineering
University of Ottawa, Ottawa, Canada
{lzymans,bouchard}@site.uottawa.ca

Abstract

The harmonic structure of the voiced speech is an effective way of conveying information in a way that is robust to white Gaussian additive noise. In this paper we propose Comb Filter Decomposition (CFD), a new method for approximating the magnitude of the speech spectrum in terms of its harmonics, which first leads to a new interpretation of the normalized autocorrelation function. Then we introduce some feature extraction methods based on CFD and on standard autocorrelation, that emphasize the harmonic peaks of the speech spectrum. The results show an improved ASR performance under noisy conditions.

1. Introduction

Most of the applications for Automatic Speech Recognition (ASR) require the system to perform in an environment with a significant amount of background noise. Present ASR technology requires a training of the system with limited amount of training data, which "teaches" the system how to recognize and categorize certain speech patterns. The background noise, present during the operation/testing stage of the ASR system, often introduces a discrepancy between the training and operation conditions, and modifies the patterns of the ASR input. These changes introduce errors in the recognition process, decreasing the ASR performance. The robust feature extraction methods attempt to identify speech characteristics that would remain similar for the clean and noisy speech signals.

The harmonic structure of the voiced speech has a remarkable resilience against additive white Gaussian noise in the way that the harmonic peaks, especially in the low frequencies, are clearly visible even with significant amount of noise in the signal. The most popular feature extraction methods, such as LPC, PLP and MFCC [1],[3], model the envelope of the magnitude of the speech spectrum, which is related to the position of the harmonics, but do not explore their potential for robustness in noisy conditions. Furthermore, all the coefficients of the feature vectors derived from the above mentioned methods are affected by changes in any part of the speech spectrum. Therefore, even when the envelope of the harmonic part of the spectrum with large peaks remains unaltered, the changes in other areas of the spectrum will induce significant alterations to the entire feature vector. PLP and MFCC tend to perform better than LPC in low Signal to Noise Ratio (SNR), because the logarithmic scaling of the frequency axis in those methods puts more emphasis on the low frequency content, which in case of voiced sounds is more resistive to corruption by additive noise. Over the years, a vast number of methods have been proposed for dealing with noisy speech [2],[4],[5], but the present ASR technology has still far to go before matching, or coming even close to, the human ability to process speech in noisy environments.

The paper is organized as follows. Section 2 starts with a short overview of the comb filters and presents the Comb Filter Decomposition, Section 3 shows the simulation results, followed by a discussion and a conclusion in Section 4.

2. Comb Filter Decomposition

2.1. Motivation

The Comb Filter Decomposition (CFD) is an attempt to analyze the speech signal in terms of its harmonic content. The motivation is to emphasize the harmonic peaks in the speech spectrum from which we can obtain feature vectors that are less sensitive to noise, especially for voiced signals.

2.2. Comb Filters

A comb filter is a filter with a frequency response consisting of a number of peaks spread equally throughout the frequency axis. The basic comb filter is a k-delay feedback comb filter with transfer function:

\[ H_k(z) = \frac{1}{1-w_k z^{-k}} \]  

(1)

This filter has a graphical representation shown in Figure 1, where \( z^k \) represents the delay of \( k \) samples, \( w_k \) is the coefficient of the delayed sample, \( u(n) \) is the input and \( s(n) \) is the output. The time domain input-output relationship of the feedback comb filter is

\[ s(n) = u(n) + w_k s(n-k) \]  

(2)

The delay \( k \) determines the number of harmonics – there are \( k \) harmonics in the frequency range \([-\pi,\pi]\) of the filter. This characteristic of the comb filter becomes apparent when looking at the location of the poles and zeros of the filter's frequency response in the \( z \)-domain [6]. There are \( k \) zeros at the origin and \( k \) regularly spread, equal-magnitude poles \( z_n \), given by:

![Figure 1: k-delay feedback comb filter](image-url)
taking the expectation of can be found from the least mean squares method [7]. Thus, setting it to zero, we can solve for \( s(n) \) where the coefficient \( w_k \) for different delays that gives the best fit of the speech spectrum. This can be accomplished by rewriting equation (2) to have

\[
E[u^2(n)] = E[s^2(n)] - 2w_1E[s(n)s(n-k)] + w_2E[s^2(n-k)]
\]  

After taking the derivative of \( E[u^2(n)] \) with respect to \( w_k \) and setting it to zero, we can solve for \( w_k \):  

\[
w_k = \frac{E[s(n)s(n-k)]}{E[s^2(n-k)]} = \frac{r_k(-k)}{r_{k-0}}
\]  

where  

\[
r_k(-k) = \sum_{s=0}^{N-1} s(n)s(n-k), \quad r_{k-0} = \sum_{s=0}^{N-1} s(n-k)^2
\]

and \( N \) is the frame length. The expectation of \( s(n)s(n-k) \) is the autocorrelation of \( s(n) \) at lag \(-k\) and the expectation of \( s(n-k)s(n-k) \) is the autocorrelation of \( s(n-k) \) at lag 0. For each frame, the CFD finds the values of \( w_k \) for \( k=1,2,\ldots,K \).

Figure 2 shows some examples of magnitude frequency response of feedback comb filters. The response of feedback comb filters for various values of delay \( k \) and coefficient \( w_k \):

\[
z_e = w_k e^{j 2\pi \frac{k}{z-1}}, \quad \text{for } w_k > 0
\]  

\[
z_e = |w_k| e^{j \frac{k}{2} \frac{2\pi}{z-1}}, \quad \text{for } w_k < 0
\]

where \( k = 1,\ldots,K \).

2.3. Comb Filter Decomposition

In the proposed Comb Filter Decomposition (CFD) the speech spectrum is modeled by a series of independent comb filters of varying delays. The goal is to determine the value of the coefficient \( w_k \) for different delays that gives the best fit of the speech spectrum. This can be accomplished by finding \( w_k \) that minimizes \( u(n) \) for a given \( k \). This value can be found from the least mean squares method [7]. Thus, taking the expectation of \( u^2(n) \) from equation (4) gives

\[
E[u^2(n)] = E[s^2(n)] - 2w_1E[s(n)s(n-k)] + w_2E[s^2(n-k)]
\]

Figure 2 shows some examples of magnitude frequency response of feedback comb filters. The response of feedback comb filters for various values of delay \( k \) and coefficient \( w_k \):

\[
z_e = w_k e^{j 2\pi \frac{k}{z-1}}, \quad \text{for } w_k > 0
\]  

\[
z_e = |w_k| e^{j \frac{k}{2} \frac{2\pi}{z-1}}, \quad \text{for } w_k < 0
\]

where \( k = 1,\ldots,K \).

2.4. Features from CFD

We propose three methods for deriving features from the CFD (these methods apply to the ACFD as well). The simplest one is forming a feature vector from the first 12 CFD coefficients for delays \( k=1,\ldots,12 \). These coefficients correspond to the first 12 lowest harmonic comb-filters, where the lower harmonics are those that have a small number of harmonic peaks in the spectrum. These low harmonics describe the general shape of the speech spectrum, which is related to the envelope of the true spectrum.

Another proposed feature extraction method is calculating the cascade frequency response of the comb filters, and then computing of a LPC analysis on the inverse FFT of the resulting composite spectrum. First, the frequency response \( H_k(m) \) for each comb filter \( k=1,\ldots,K \) is calculated by substituting \( z = e^{j 2\pi \frac{m}{M}} \) in equation (1) for
speech data was downsampled to 8kHz and divided into 20ms end of sentences from the TIMIT database were used. The single word utterances without silence at the beginning or the end for the probability distribution model of the feature vectors.

All the simulations were done using Matlab 7.0. The ASR calculations are done as specified in the ITU-T G.729 codec. A division by \( K \) is also applied to normalize the CFD cascade spectrum. The resulting formula follows:

\[
H_k(m) = H_k(e^{\frac{j2\pi m}{K}})
\]  

(7)

Then, the magnitudes of the frequency response of all the comb filters are multiplied to form the CFD cascade spectrum. The logarithm is then taken to compensate for the exaggeration of the harmonic peaks due to the previous multiplication. A division by \( K \) is also applied to normalize the CFD cascade spectrum. The resulting formula follows:

\[
|H(m)| = \frac{1}{K} \ln \prod_{k=1}^{K} |H_k(m)|
\]  

(8)

The third feature extraction method follows the same steps as the previous method with the addition of LSF conversion after the LPC analysis. It is well known that LSF coefficients show a better quantization and classification performance than LPC coefficients [11]. The LSF calculations are done as specified in the ITU-T G.729 codec.

3. Simulation Results

All the simulations were done using Matlab 7.0. The ASR system was built using an HMM classifier with 5-mix GMM for the probability distribution model of the feature vectors. Single word utterances without silence at the beginning or the end of sentences from the TIMIT database were used. The speech data was downsampled to 8kHz and divided into 20ms frames with 50% overlap. A Hamming window was applied to the speech frames, with the exception of the CFD and the ACFD computations. Noisy conditions were simulated by adding white Gaussian noise at different SNRs to the speech data.

The recognition was performed on a 10-word dictionary consisting of the words: suit, like, greasy, oily, dark, year, rag, ask, wash, don’t. 10 HMMs were created, one for each word class, with the states corresponding to the phonemes in the phoneme alphabet of the above mentioned 10-word dictionary. The training of each HMM was done by running the EM algorithm using 20 utterances of its corresponding word class, with the phoneme label for each frame of speech being available. Testing was done with 100 different utterances of each word by running the Viterbi algorithm on each HMM to find the best state sequence for features from a given test utterance. The word class of the test utterance was declared to be the one corresponding to the HMM that returned the highest probability of its best state sequence.

The feature extraction methods tested were: LPC – 12 LPC coefficients, MFCC - 7 coefficient MFCC from 30 Mel-bank filters, CFD – 12 CFD coefficients, CFD+LPC – 12 LPC coefficients from 160 CFD cascade spectrum, CFD+LSF – LSF on 12 CFD coefficients from 160 CFD cascade spectrum, ACFD – 12 ACFD coefficients, ACFD+LPC – 12 LPC coefficients from 160 ACFD cascade spectrum, ACFD+LSF – LSF on 12 CFD coefficients from 160 ACFD cascade spectrum. Other robust feature methods used for comparison were: PLP+LPC – 12 coefficient LPC on a PLP preprocessed signal [3], MFCC+CMS+CN – 7 normalized MFCC coefficients from 30 Mel-bank filters [4], SENT+D – 5 coefficient multi-band spectral entropy [5] with delta
features (it was observed that spectral entropy based feature extraction performs better with delta features than with plain features). Figure 5 and Table 1 show the simulation results. The simulation results show that CFD based features give a significant improvement of the recognition rates in low SNR, while matching the LPC method in the high SNR scenarios. The CFD and the ACFD have almost the same performance, while the CFD+LSF and ACFD+LSF are far more resistant to noise than any other method. In low SNR the recognition error for CFD+LSF is minimally larger than for ACFD+LSF, but overall it gives a better performance. Spectral entropy with delta features gives the best recognition rates on the clean speech, while matching the CFD and ACFD methods in the noisy conditions.

### 4. Discussion and Conclusions

A new method for robust feature extraction based on harmonic decomposition of the speech signal was presented. The results show that the CFD-based features improve the performance of ASR systems in environments with background noise that has white Gaussian characteristics.

The computational complexity of the CFD when the number of samples in the frame \( N \) is significantly larger than the number of coefficients \( K \) is of the order \( \sim O(N^2) \), which is more complex than the LPC under the same conditions. On the other hand, the ACFD has a complexity of the order \( \sim O(N) \). This is less complex than the LPC, because all that is required is the autocorrelation of the speech frame without the Levinson-Durbin calculation. When \( K=N \), as it is in the case of the CFD+LPC method, the complexity grows to \( \sim O(N^3) \), with additional calculations needed to derive the cascade spectrum, the IFFT and the LPC coefficients, it becomes computationally intensive. The ACFD+LPC complexity for \( K=N \) is \( \sim O(N^2) \). Some additional complexity (LSF conversion) is required for CFD+LSF and ACFD+LSF.

Further work on the CFD could include other feature extraction methods from CFD coefficients - for instance, a method that would place a larger emphasis on the low frequencies of the speech spectrum. Rather than using the comb filter with a constant peak magnitude, it should be possible to shape the frequency response of the comb-filter to match the envelope of the speech spectrum in order to have a better fit to the spectrum content. This should improve the recognition accuracy in both the clean and noisy conditions.

### 5. Acknowledgements

The authors would like to thank SoftdB (http://www.softdb.com) for funding this research.

### 6. References


