Speech Feature Compensation Based on Pseudo Stereo Codebooks for Robust Speech Recognition in Additive Noise Environments

Tsung-hsueh Hsieh and Jeih-weih Hung

Department of Electrical Engineering, National Chi Nan University, Taiwan
s94323532@ncnu.edu.tw, jwhung@ncnu.edu.tw

Abstract

In this paper, we propose several compensation approaches to alleviate the effect of additive noise on speech features for speech recognition. These approaches are simple yet efficient noise reduction techniques that use online constructed pseudo stereo codebooks to evaluate the statistics in both clean and noisy environments. The process yields transforms for noise-corrupted speech features to make them closer to their clean counterparts. We apply these compensation approaches on various well-known speech features, including mel-frequency cepstral coefficients (MFCC), autocorrelation melfrequency cepstral coefficients (AMFCC) and perceptual linear prediction cepstral coefficients (PLPCC). Experimental results conducted on the Aurora-2 database show that the proposed approaches provide all types of the features with significant performance gain when compared to the baseline results and those obtained by using the conventional utterance-based cepstral mean and variance normalization (CMVN).

Index Terms: cepstral statistic compensation, MFCC, AMFCC, PLPCC, polynomial regression

1. Introduction

The performance of a speech recognition system is often severely degraded in the presence of noise. A variety of approaches have been proposed to alleviate the effect of additive noise. They can be roughly divided into three classes: utilization of a noise robust representation of speech signals, enhancement of the speech features before they are fed to the recognizer, and adaptation of the speech models in the recognizer to make them better match the noise conditions. The main difference between the first two classes of approaches is that, for the first class, the noise robust speech features are used for both model training and testing, and for the second, enhancement procedures are often performed only on the testing noise-corrupted speech, while keep the speech features for training unchanged. In this paper, our proposed approaches belong to the second class.

Among the first class of approaches, cepstral mean subtraction (CMS) [1] and cepstral mean and variance normalization (CMVN) [2] are two well-known algorithms with which the statistics of the cepstral features, that is, the mean or the mean and the variance, are normalized in order to reduce the mismatch between the training and testing conditions. In conventional CMS and CMVN, the statistics of the features are often evaluated from the entire frame set of an utterance. While these utterance-based CMN and CMVN techniques are simple to implement, they likely have some inherent drawbacks. First, they cannot be implemented in an on-line manner since the computation and normalization of the statistics cannot be performed until the last frame of an utterance is received. Secondly, the number of frames in an utterance, which is often an uncontrollable factor, influences the accuracy of the obtained statistics. Thirdly, because the length, or the total number of different acoustic units, may vary from utterance to utterance, the normalized features of the same acoustic unit in an utterance may differ from those in another utterance.

In our previous work [3], we proposed the approach of cepstral statistics compensation (CSC) and some of its variants, and showed that they brought improvement for the mel-frequency cepstral coefficients (MFCC) when compared with the utterance-based CMVN. Furthermore, they can be implemented in an almost on-line manner. In this paper, we attempt to apply these CSC-related approaches to some other types of speech features to see if similar improvements can be obtained. Besides MFCC, two other well-known types of speech features are used here, including autocorrelation mel-frequency cepstral coefficients (AMFCC) [4], and perceptual linear prediction cepstral coefficients (PLPCC) [5]. The CSC-related approaches are mainly based on two sets of codewords, named "pseudo stereo codebooks", in which one is for clean speech condition, and the other for the noise-corrupted speech condition. The noise-corrupted speech codebook is constructed by properly integrating the clean speech codebook and the noise estimates. More clearly, during the feature extraction processes, we find an intermediate feature domain in which the clean speech and noise are linearly additive. This intermediate feature may be the magnitude spectrum for MFCC and AMFCC, or the autocorrelation coefficients and the magnitude spectrum for PLPCC. The clean speech codewords for the intermediate feature domain are first constructed, and they are then linearly added with the noise estimates to compose the noise-corrupted speech codewords for that domain. Finally, they are transformed to the final feature domain following the remaining feature extraction processes.

With the pseudo stereo codebooks in hand, we develop several approaches in order to reduce the mismatch between the training and testing conditions. For the approach of CSC, the statistics (mean and variance) obtained from the codebooks are used to find a transformation for testing speech features that will make them closer to clean speech features in statistics. Polynomial regressions, such as linear least squares (LLS) and quadratic least squares (QLS) regressions, are also used as the transformations for testing speech features in order to minimize the overall pair-wise squared distances between the two sets of codewords. The remainder of the paper is organized as follows: section 2 presents the construction of pseudo stereo codebooks for MFCC, AMFCC and PLPCC, respectively. Section 3 introduces several codebook-based feature compensation approaches. The experimental environment setup is described in section 4, and the recognition results are given and discussed in section 5. Finally, section 6 contains brief conclusions.
2. The Construction of Pseudo Stereo Codebooks

In this section, we introduce the approach to constructing codebooks for clean training and noise-corrupted testing environments, respectively. The main idea of this approach is based on the assumption that the noise-corrupted speech is approximately the linear sum of clean speech and noise in a specific feature domain. Here, three types of speech features are considered, and they are MFCC, AMFCC and PLPCC. The derivation processes of them are shown in Figures 1 (a)-(c). Although it is for different types of features, the approach to constructing the pseudo stereo codebooks is quite unified. Following the derivation processes of the speech feature for a speech signal, we first convert each clean speech utterance in the clean training database into a sequence of intermediate feature vectors. Note that in this intermediate feature domain, the noise-corrupted speech is approximately the linear sum of the clean speech and noise. For MFCC and AMFCC the intermediate features are the magnitude spectrum, and for PLPCC, they are the magnitude spectrum or the correlation coefficients. These vectors, taking from all of the utterances in the training set, are then used to construct a set of $N$ codewords by vector quantization (VQ), denoted as $\{x[n], 1 \leq n \leq N\}$. These codewords are then transformed into the cepstral domain following the remaining feature extraction procedures as follows,

$$x[n] = f(x[n]),$$  

where $f(\cdot)$ denotes the transformation function and it depends on the chosen feature type. Thus the set of codewords $\{x[n], 1 \leq n \leq N\}$ is the clean speech cepstral codebook. Note that we construct this cepstral codebook by transforming the intermediate-feature codewords rather than by vector quantizing the cepstral features directly, and that these intermediate-feature codewords are preserved in order to construct the noise-corrupted speech codebook.

For the noise-corrupted testing environment, however, it is often difficult to obtain a set of reliable codewords completely based on a single testing utterance. As a result, we attempt to construct this cepstral codebook by transforming the intermediate-feature codewords rather than by vector quantizing the cepstral features directly, and that these intermediate-feature codewords are preserved in order to construct the noise-corrupted speech codebook.

For a given noise-corrupted testing utterance, let the estimated noise in the intermediate feature domain be approximated as a set of vectors, which are denoted as $\{\tilde{u}[p], 1 \leq p \leq P\}$. Then, since the clean speech and noise are approximately additive in this intermediate feature domain, the noise-corrupted speech codewords are obtained as

$$\tilde{y}[m]|_{n=(n-l)p+p} = x[n] + \tilde{u}[p].$$  

Finally, we transform each $\tilde{y}[m]$ into the cepstral domain as in eq. (1),

$$y[m] = f(\tilde{y}[m]).$$

Thus the set of codewords $\{y[m], 1 \leq m \leq NP\}$ is the noise-corrupted speech cepstral codebook. From the above, the two sets of codewords, $\{x[n]\}$ and $\{y[m]\}$, are viewed as the representatives for the clean training and noise-corrupted testing conditions, respectively, and they are named "pseudo stereo codebooks" here. The term "pseudo" indicates that the noise-corrupted speech codebook is not derived from the noise-corrupted speech directly, but is a fusion of the clean speech codebook and the noise estimates. In addition, note that we construct the clean speech codebook $\{x[n]\}$ just once with all clean utterances in the training database, and that this is done in an off-line manner. However, the noise-corrupted speech codebook $\{y[m]\}$ needs to be updated for each different testing utterance or when the noise condition is altered. Since the noise estimates $\{\tilde{u}[p]\}$ can often be obtained with the several leading frames of an utterance, the noise-corrupted speech codebook $\{y[m]\}$ can be constructed in an almost on-line manner without a long delay.

Figure 1. The derivation processes of the speech features (a) MFCC (b) AMFCC (c) PLPCC

3. The Robustness Approaches Based on the Pseudo Stereo Codebooks

The pseudo stereo codebooks provide us with the information about the speech features under clean training and noisy
testing environments. Based on the pseudo stereo codebooks, in the following subsections we introduce two series of approaches that attempt to reduce the effect of additive noise.

3.1. Cepstral Statistics Compensation

The pseudo stereo codebooks help us obtain the approximate statistics for the features of the clean and noise-corrupted speech. For example,

\[ \mu_{x_i} \approx \frac{1}{N} \sum_{n=1}^{N} (x[n])_i, \quad \sigma_{x_i}^2 \approx \frac{1}{N} \sum_{n=1}^{N} [(x[n])_i - \mu_{x_i}]^2, \]

\[ \mu_{y_i} \approx \frac{1}{NP} \sum_{n=1}^{NP} (y[m])_i, \quad \sigma_{y_i}^2 \approx \frac{1}{NP} \sum_{n=1}^{NP} [(y[m])_i - \mu_{y_i}]^2, \]

where \( \mu_{y_i} \) denotes the \( i \)-th component of an arbitrary vector \( y \), \( \mu_{x_i} \) and \( \sigma_{x_i}^2 \) are the mean and variance of the \( i \)-th component of the clean speech feature vector \( x \), respectively, and \( \mu_{y_i} \) and \( \sigma_{y_i}^2 \) are the mean and variance of the \( i \)-th component of the noise-corrupted speech feature vector \( y \), respectively. Note that in eqs. (4) and (5), the statistics are obtained by averaging the two sets of codewords, \( \{x[n]\} \) and \( \{y[m]\} \), respectively. Given these statistics, in the algorithm of cepstral statistics compensation (CSC) we transform each noise-corrupted cepstral vector \( y \) as

\[ (z)_i = \frac{\sigma_{x_i}}{\sigma_{y_i}} \times [(y)_i - \mu_{y_i}] + \mu_{x_i}, \]

where the parameters \( \mu_{x_i} \), \( \mu_{y_i} \), \( \sigma_{x_i} \), and \( \sigma_{y_i} \) are obtained by eqs. (4) and (5). Ideally, the new feature \( (z)_i \) and the clean speech feature \( (x)_i \) will have the same mean and variance if eq. (6) is used. Since some of the statistics (mean and variance here) for the noise-corrupted speech cepstra \( y \) are compensated in order to approximate those of the clean speech cepstra \( x \), eq. (6) is called cepstral statistics compensation (CSC) algorithm. The concept of CSC is similar to the conventional utterance-based cepstral mean and variance normalization (U-CMVN) because they pursue to generate similar statistics for the training and testing cepstra. However, CSC offers some advantages. First, it can be implemented in an almost on-line manner since the clean speech codebook has been created in advance, and the noise estimates for constructing the noise-corrupted speech codebook often can be obtained with the first several frames in a testing utterance. Secondly, in CSC the statistics are obtained indirectly from the information contained by all the utterances in the training database, while U-CMVN just employs a single utterance to determine the means and the variances. Therefore it is expected that more accurate estimates of these statistics can be achieved with the codebooks.

3.2. Polynomial Regression Approaches — Linear Least Squares and Quadratic Least Squares Regressions

Section 2 explains that each noise-corrupted speech codeword \( y[m] \) corresponds to its clean counterpart \( x[n] \), where \( n = \lfloor m/P \rfloor \) (\( \lfloor \cdot \rfloor \) denotes the "ceil" operation), and the two sets of codewords \( \{x[n]\} \) and \( \{y[m]\} \) are assumed to represent the clean and noise-corrupted speech cepstra \( x \) and \( y \), respectively. If we can find a transformation \( T(\cdot) \) for each \( y[m] \) that minimizes the overall distances between \( T(y[m]) \) and \( x[n] \), or their variations, then it can be reasonably expected to yield a \( T(y) \) very close to its clean version \( x \) when this transformation is performed on an arbitrary noise-corrupted speech cepstrum \( y \). For simplicity, the transformation considered here is component-wise, that is, it is performed on each component of \( y \). Let \( T_i(\cdot) \) be the transformation function for the \( i \)-th component of \( y \), and the objective function to be minimized is the overall squared distances,

\[ J_i = \sum_{n=1}^{N} \| T_i((y[m])_i) - (x[n])_i \|^2, \]

where \( n = \lfloor m/P \rfloor \). In particular, if \( T_i(\cdot) \) is a polynomial of degree \( K \), then minimizing \( J_i \) in eq. (7) with respect to \( T_i(\cdot) \) becomes a classical least-squares (LS) problem [6]. That is, if

\[ T_i(u) = a_0^i + a_1^i u + a_2^i u^2 + \cdots + a_K^i, \]

and the objective function in eq. (8) is re-written in vector-matrix form as

\[ J_i = \| Y_i a_i - b_i \|^2, \]

where the \( m,n \)-th entry of the matrix \( Y_i \) is

\[ (Y_i)_{mn} = [(y[m])_i]^{mn+1}, \quad 1 \leq m \leq NP, \quad 1 \leq n \leq K + 1, \]

\[ a_i = \begin{bmatrix} a_0^i & a_1^i & \cdots & a_K^i \end{bmatrix}^T, \]

and

\[ b_i = \begin{bmatrix} (x[[y]])_i \quad (x[[y]])_i \quad \cdots \quad (x[[y]])_i \end{bmatrix}^T, \]

then the coefficient vector \( a_i \) of the polynomial \( T_i(\cdot) \) that minimizes \( J_i \) is the following least-squares solution,

\[ a_i = (Y_i^T Y_i)^{-1} Y_i^T b_i. \]

Practically, the degree \( K \) of the polynomial \( T_i(\cdot) \) is not set too large to prevent over-fitting or the ill-conditional matrix \( Y_i^T Y_i \). Here we only consider two cases: if \( K = 1 \), the transformation \( T_i(\cdot) \) is a linear function, and is often called a linear regression (LR) or linear least squares (LLS) regression; if \( K = 2 \), \( T_i(\cdot) \) is a quadratic function and is called a quadratic least squares (QLS) regression.

4. Experimental Setup

The proposed codebook-based algorithms have been tested with the AURORA-2 database. For the recognition experiments, two sets (Test Sets A and B) of utterances artificially contaminated by different types of noise (subway, babble, car, etc.) under different SNR levels (ranging from -5dB to 20dB) are prepared. Since the proposed algorithms only involve the front-end feature extraction, the procedures for training and recognition are identical to the reference experiments stated in the AURORA-2 documentation [7]. For the clean training database, each of the 8440 strings is first converted into a stream of intermediate feature vectors. For MFCC and AMFCC, the chosen intermediate features are
the 23-dimensional mel-spectrum. For PLPCC, they are 40 correlation coefficients or 40-dimensional magnitude spectrum. All of these intermediate feature vectors are used to construct a set of $N$ codewords via vector quantization (VQ). (Note that since the number of autocorrelation coefficients used in the derivation of AMFCC is relatively large, which makes the VQ processes difficult, we do not choose the correlation coefficients as the intermediate features for AMFCC). These codewords are then converted to 13-dimensional cepstral vectors to form the clean speech cepstral codebook $\{x[m]\}$. The size $N$ of the codebook is set to give the best recognition performance. In addition, all of these intermediate feature vectors in the training set are converted to cepstral domain. The obtained cepstral features plus their delta and delta-delta are the components in the finally used 39-dimensional feature vectors. These feature vectors are then used to train the hidden Markov models for each digit.

For the testing condition, the leading five frames of intermediate features for each utterance are used as the estimated noise vectors. Then, following the procedures stated in section 2, we constructed the noise corrupted speech cepstral codebook $\{y[m]\}$. Then based on the two codebook $\{x[m]\}$ and $\{y[m]\}$, the various proposed algorithms are performed to adjust the testing features, respectively.

### 5. Experimental Results and Discussions

Table 1 lists the averaged recognition results (0-20dB) of the baseline and several robustness approaches, utterance-based CMVN (U-CMVN), the proposed cepstral statistics compensation (CSC), linear least squares (LLS) and quadratic least squares (QLS) for various types of speech features including MFCC, AMFCC and PLPCC. From this table, several phenomena can be observed:

1. When no compensation approach is performed, the baseline results for the three types of speech features show that AMFCC performs the best for both Test Sets A and B. MFCC performs better than PLPCC for Test Set A, but both of them achieve very similar results for Test Set B.
2. With conventional U-CMVN, the performance improvement is obvious for all three types of features, and the improvement is particularly significant for PLPCC.
3. The proposed CSC enhances the noise robustness of the original speech features significantly. In addition, CSC behaves significantly better than U-CMVN for all types of features under all noise conditions.
4. The two polynomial regression approaches also improve the recognition accuracy significantly. In most cases LLS and QLS perform similarly to CSC, and better than U-CMVN (with QLS using the correlation coefficients as the intermediate feature being the only exception).
5. With PLPCC, choosing the magnitude spectrum as the intermediate feature rather than the correlation coefficients achieves a better recognition performance for LLS and QLS. However, the two types of intermediate features result in similar recognition accuracy for CSC.

In summary, compared with the baseline and U-CMVN, the proposed three approaches, CSC, LLS, and QLS, produce significant recognition improvements for all three types of speech features discussed here. In addition, with these approaches the performance differences among MFCC, AMFCC and PLPCC become less significant. That is, similar excellent recognition accuracy can be achieved with these approaches regardless of what type of speech feature we use.

### 6. Conclusions

In this paper, we design the pseudo stereo codebooks that respectively represent the clean and noisy conditions. With these codebooks some novel noise robustness algorithms are proposed, including cepstral statistics compensation (CSC), linear least squares (LLS) and quadratic least squares (QLS). We show that all of them are capable of improving the recognition performance of various types of speech features, including the well-known MFCC, AMFCC and PLPCC, in a noise corrupted environment. Compared with conventional utterance-based cepstral mean and variance normalization, these new approaches can be implemented in an on-line manner, and they provide better noise robustness.

### 7. References


<table>
<thead>
<tr>
<th>Test Set</th>
<th>System</th>
<th>MFCC</th>
<th>AMFCC</th>
<th>PLPCC (1)</th>
<th>PLPCC (2)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>baseline</td>
<td>61.13</td>
<td>65.52</td>
<td>57.38</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>U-CMVN</td>
<td>70.38</td>
<td>68.78</td>
<td>74.48</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CSC</td>
<td>78.00</td>
<td>78.02</td>
<td>77.87</td>
<td>77.18</td>
</tr>
<tr>
<td></td>
<td>LLS</td>
<td>77.11</td>
<td>76.46</td>
<td>75.78</td>
<td>78.12</td>
</tr>
<tr>
<td></td>
<td>QLS</td>
<td>78.52</td>
<td>75.41</td>
<td>72.82</td>
<td>77.31</td>
</tr>
<tr>
<td></td>
<td>baseline</td>
<td>55.57</td>
<td>59.43</td>
<td>54.51</td>
<td></td>
</tr>
<tr>
<td></td>
<td>U-CMVN</td>
<td>71.10</td>
<td>71.93</td>
<td>75.63</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>CSC</td>
<td>76.05</td>
<td>77.22</td>
<td>77.95</td>
<td>77.66</td>
</tr>
<tr>
<td></td>
<td>LLS</td>
<td>75.59</td>
<td>76.56</td>
<td>76.70</td>
<td>79.51</td>
</tr>
<tr>
<td></td>
<td>QLS</td>
<td>76.97</td>
<td>76.91</td>
<td>73.41</td>
<td>77.90</td>
</tr>
</tbody>
</table>

Table 1. Recognition accuracy (%) averaged over 5 SNR values, 0-20dB, for the baseline and various approaches, U-CMVN, CSC, LLS and QLS, of three different types of speech features, MFCC, AMFCC and PLPCC under Test Sets A and B of Aurora 2 database. Note that we use two different intermediate features for PLPCC, which are (1) autocorrelation coefficients (2) magnitude spectrum.