Cepstral Statistics Compensation Using Online Pseudo Stereo Codebooks for Robust Speech Recognition in Additive Noise Environments

Jeih-weih Hung

Dept of Electrical Engineering, National Chi Nan University
Taiwan, Republic of China

email : jwhung@ncnu.edu.tw

Abstract

In this paper, we propose the cepstral statistics compensation (CSC) algorithm, which alleviates the effect of additive noise on the cepstral features for speech recognition. It is a simple but quite efficient noise reduction technique that makes use of online constructed pseudo stereo codebooks. The statistics, such as mean and variance, for the cepstral features in both clean and noisy environments are evaluated using pseudo stereo codebooks. Then a transform is obtained for the noise-corrupted cepstra so that the statistics of the transformed ones are close to those of clean cepstra. Experimental results show that CSC provided a 13% reduction in word error rate when compared to the results obtained using cepstral mean and variance normalization (CMVN), and a 34% reduction in error rate compared to baseline processing in the noise range of 0-20dB in experiments conducted on Aurora-2 Test Set A noisy digits database. In addition, we also provide some other noise robustness approaches based on pseudo stereo codebooks and show their effectiveness in noisy speech recognition.

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 in order to make them better match the noisy 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 keeping the speech features for training unchanged. In this paper, our proposed approaches belong to the second class. A new feature enhancement pre-processing scheme called cepstral statistics compensation (CSC) is introduced.

The philosophy of the CSC approach can be summarized as follows. Due to the presence of noise, the statistics, for example, mean and variance, of the resulted noise corrupted speech features are quite different from those the original clean ones. If the statistics of both clean and noise-corrupted speech features can be obtained, or approximately estimated, then we are able to transform the noise corrupted speech features in order to make them similar to clean ones in their statistics.

First, in order to efficiently obtain the statistics of clean speech features, we collect all clean speech features (in *mel-spectral* domain) in the training database and create a "clean-speech" mel-spectral codebook of N codewords via vector quantization (VQ). These mel-spectral codewords are then transformed into cepstral domain. Viewing these cepstral codewords as samples, we can calculate their statistics, for example, the mean, variance or higher-order moments. We assume these obtained statistics are close to the exact ones of all clean speech features in the training database.

Next, for the noisy testing environment, however, since a testing utterance is often short in length and the signal-to-noise ratio (SNR) is often time-varying from utterance to utterance, it is often difficult to obtain a set of reliable codewords with which the statistics for noise-corrupted speech are estimated. As a result, here we attempt to construct the "noise-corrupted speech" codebook with the help of the available "clean-speech" codebook. For a given testing noisy utterance, the noise-only part is first detected and then a noise vector in *mel-spectral* domain is estimated, which approximates the noise level of this utterance. Then this noise vector is linearly added to every *mel-spectral* "clean-speech" codeword to form a set of *mel-spectral* "noise-corrupted speech" codewords. Similar to the procedures for clean speech, these mel-spectral codewords are transformed into cepstral domain. With these cepstral codewords, the approximated statistics of the noise-corrupted speech cepstra can be estimated.

Finally, with the statistics for both conditions in hand, a transformation for the testing noise-corrupted speech cepstra can be obtained so that the new statistics of the transformed cepstra can be equal or close to those of clean cepstra. We believe that such a transformation is capable of reducing the mismatch between clean training and noisy testing conditions, and thus improve the robustness of the speech recognition system. The advantage of this CSC algorithm is that it is very simple since only the testing data is processed, while the training data and the recognition models remain unchanged. Also, because the noise information can be often extracted in the first few frames of an utterance, it can be on-line performed. Experimental results show that, with simple noise estimation, the proposed CSC algorithm can significantly improve the recognition accuracy of the original MFCC features under an additive noise environment. Moreover, it was also shown that CSC outperforms the widely used cepstra mean and variance normalization (CMVN) approach [1].

Besides the above CSC algorithm, the two sets of codewords, which we call "pseudo stereo codebooks" afterwards, still have extensive applications. Several other noise robustness approaches based on them can be easily developed. For example, the codebook-based cepstral mean normalization (CMN) and CMVN approaches use the statistics obtained from these codebooks rather than from per utterance. In addition, in order to minimize the overall pair-wise square distances between the two sets of codewords, polynomial regressions, such as linear least square and quadratic least square regression, can be used as the transformations for the testing speech cepstra. Experimental results also reveal that these codebook-based approaches are very effective in reducing the effect of additive noise by increasing the recognition accuracy.

This paper is organized as follows: In section 2, the construction

of pseudo stereo codebooks is stated and the proposed CSC algorithm is formally derived. Section 3 summarizes several other noise robustness approaches based on pseudo stereo codebooks. The experimental environment setup is described in section 4, and the recognition results are given and discussed in section 5. Section 6 compares the proposed algorithms with some other existing approaches. Finally, section 7 briefly presents conclusions and future works.

2. Pseudo Stereo Codebooks and Cepstral Statistics Compensation

Given the clean training database, we first convert each utterance into a sequence of mel-spectral vectors. All of these mel-spectral vectors are then used to construct a set of N codewords, denoted as $\{\tilde{\mathbf{x}}_m,\ 1 \le m \le N\}$, (the vectors in mel-spectral domain are indicated by the notation "~" here). These mel-spectral codewords can be transformed into cepstral domain as follows,

$$\mathbf{x}_m = \mathbf{C}\log(\tilde{\mathbf{x}}_m), \tag{1}$$

where $\mathbf{C}$ is the DCT matrix.

Under noisy testing conditions, let the estimated mel-spectrum of the noise be just approximated as a vector $\tilde{\mathbf{n}}$ for simplicity. Then, since the clean speech and noise are roughly additive in mel-spectral domain, the noise-corrupted speech codewords $\{\tilde{\mathbf{y}}_m,\ 1 \le m \le N\}$ are obtained as

$$\tilde{\mathbf{y}}_m = \tilde{\mathbf{x}}_m + \tilde{\mathbf{n}}. \tag{2}$$

Finally, we transform each $\tilde{\mathbf{y}}_m$ into cepstral domain as in eq. (1),

$$\mathbf{y}_m = \mathbf{C}\log(\tilde{\mathbf{y}}_m) \tag{3}$$

From the above, the two set of codewords, $\{\mathbf{x}_m\}$ and $\{\mathbf{y}_m\}$ can be viewed as the cepstral codebooks for the clean training and noisy testing conditions, respectively, and they are named "pseudo stereo codebooks" here.

The pseudo sterero codebooks may help us obtain the approximate statistics for the cepstra of the clean and noise corrupted speech. For example,

$$\mu_{\mathbf{x},i} = \frac{1}{N}\sum_{m=1}^{N}(\mathbf{x}_m)_i,\ \sigma^2_{\mathbf{x},i} = \frac{1}{N}\sum_{m=1}^{N}\left[(\mathbf{x}_m)_i - \mu_{\mathbf{x},i}\right]^2,$$
$$\mu_{\mathbf{y},i} = \frac{1}{N}\sum_{m=1}^{N}(\mathbf{y}_m)_i,\ \sigma^2_{\mathbf{y},i} = \frac{1}{N}\sum_{m=1}^{N}\left[(\mathbf{y}_m)_i - \mu_{\mathbf{y},i}\right]^2, \tag{4}$$

where $(\mathbf{v})_i$ denotes the i-th component of an arbitrary vector $\mathbf{v}$, $\mu_{\mathbf{x},i}$ and $\sigma^2_{\mathbf{x},i}$ are the mean and variance of the i-th component of the clean speech cepstral vector $\mathbf{x}$, respectively, and $\mu_{\mathbf{y},i}$ and $\sigma^2_{\mathbf{y},i}$ are the mean and variance of the i-th component of the noise corrupted speech cepstral vector $\mathbf{y}$, respectively. As a result, we can transform each noise corrupted cepstral vector $\mathbf{y}$ as

$$(\mathbf{z})_i = (\mathbf{y})_i - \mu_{\mathbf{y},i} + \mu_{\mathbf{x},i}, \tag{5}$$

or

$$(\mathbf{z})_i = \frac{\sigma_{\mathbf{x},i}}{\sigma_{\mathbf{y},i}} \times \left[(\mathbf{y})_i - \mu_{\mathbf{y},i}\right] + \mu_{\mathbf{x},i}, \tag{6}$$

The new cepstral coefficient $(\mathbf{z})_i$ and the clean one $(\mathbf{x})_i$ will have the same mean if eq. (5) is used, or they have the same mean and variance if eq. (6) is used. Since some of the statistics (mean, or mean and variance here) of the noise corrupted speech cepstra are compensated, they can be equal or close to those of clean ones, and thus eq. (5) and eq. (6) are called cepstral statistics compensation (CSC) algorithms.

The concept of CSC is quite similar to that of the well-known cepstral smoothing techniques, cepstral mean normalization [2] (CMN) and cepstral mean and variance normalization (CMVN) [1], since all of them pursue the same statistics for the training and testing speech cepstra. However, there are two major differences between CSC and these two normalization approaches. First, in CSC, only testing speech cepstra are adjusted and training speech ones remain unchanged, while in CMN and CMVN both training and testing sets need to be normalized to have zero mean, or have both zero mean and unit variance. When the signal-to-noise ratio (SNR) is high or medium, the testing MFCC features will not be altered too much by CSC, and still keep their original discriminating capability. So we expect that CSC will outperform CMN and CMVN under moderate noisy conditions. Secondly, in CSC, the statistics of the noise corrupted cepstra were estimated by the online constructed codebook, while in CMN or CMVN, the statistics are often obtained with the whole frames of an utterance or a running window on them. Therefore, compared with CMN and CMVN, CSC is more likely to perform in a real-time manner as long as the noise estimate is also real-time. Furthermore, the accuracy of the estimated statistics in CSC relies on the accuracy of the noise estimate and the level of representation of the codebook, while in CMN and CMVN, it mainly depends on the selected frames to be averaged.

3. The Other Noise Robustness Approaches Based on Pseudo Stereo Codebooks

Besides the proposed CSC approach, there are still several other approaches that can be easily developed using pseudo stereo codebooks. We briefly introduce them in the following subsections.

3.1 Codebook-based CMN and CMVN

As introduced in section 2, the widely used CMN or CMVN often uses the whole utterance frames or a running window on them to calculate the statistics. Here in codebook-based CMN and CMVN, the statistics are obtained from the pseudo stereo codebooks. Therefore, for codebook-based CMN,

$$(\underline{\mathbf{x}})_i = (\mathbf{x})_i - \mu_{\mathbf{x},i}\ ,\ (\underline{\mathbf{y}})_i = (\mathbf{y})_i - \mu_{\mathbf{y},i}, \tag{7}$$

and for codebook-based CMVN,

$$(\underline{\mathbf{x}})_i = \frac{(\mathbf{x})_i - \mu_{\mathbf{x},i}}{\sigma_{\mathbf{x},i}}\ ,\ (\underline{\mathbf{y}})_i = \frac{(\mathbf{y})_i - \mu_{\mathbf{y},i}}{\sigma_{\mathbf{y},i}}, \tag{8}$$

where $\underline{\mathbf{x}}$ and $\underline{\mathbf{y}}$ are the normalized version of training and testing cepstra, respectively, and the statistics $\mu_{\mathbf{x},i}$, $\mu_{\mathbf{y},i}$, $\sigma_{\mathbf{x},i}$ and $\sigma_{\mathbf{y},i}$ are obtained by eq. (4). Thus, ideally both $(\underline{\mathbf{x}})_i$ and $(\underline{\mathbf{y}})_i$ are zero-mean, or zero-mean and unit-variance.

3.2 Polynomial Regression Approaches–Linear Least Square and Quadratic Least Square Regressions

From section 2, it is observed that each noise corrupted speech codeword $\mathbf{y}_m$ corresponds to its clean version $\mathbf{x}_m$, and the two sets $\{\mathbf{x}_m\}$ and $\{\mathbf{y}_m\}$ are assumed to represent all the

clean and noise corrupted speech cepstra $\mathbf{x}$ and $\mathbf{y}$, respectively. If we can find a transformation $\mathcal{T}(.)$ on each $\mathbf{y}_m$ such that the overall distances, or its variations, between $\mathcal{T}(\mathbf{y}_m)$ and $\mathbf{x}_m$ are minimum, then it is reasonable to guess that, when the transformation $\mathcal{T}(.)$ is performed on an arbitrary noise corrupted speech cepstrum $\mathbf{y}$, $\mathcal{T}(\mathbf{y})$ will be very close to it clean version $\mathbf{x}$. For simplicity, the transformation considered here is component-wise, that is, it is performed on each dimension of $\mathbf{y}_m$, and the objective function to be minimized is the overall squared distances:

$$J_i = \sum_{m=1}^{N} \left[\mathcal{T}_i\left((\mathbf{y}_m)_i \right) - (\mathbf{x}_m)_i \right]^2 . \quad (9)$$

When the transformation function $\mathcal{T}_i(u)$ in eq. (9) is assumed to be a polynomial of the variable u, minimizing J_i with respect to $\mathcal{T}_i(.)$ becomes a classical least-square (LS), or curve- fitting problem. That is, if

$$\mathcal{T}_i(u) = a_n u^n + a_{n-1} u^{n-1} + \cdots + a_0 , \quad (10)$$

and the objective function J_i in eq. (9) can be re-written as

$$J_i = \left\| \mathbf{Y}\mathbf{a} - \mathbf{b} \right\|^2 , \quad (11)$$

where $\mathbf{Y} = \begin{bmatrix} (\mathbf{y}_1)_i^n & (\mathbf{y}_1)_i^{n-1} & \cdots & 1 \\ (\mathbf{y}_2)_i^n & (\mathbf{y}_2)_i^{n-1} & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ (\mathbf{y}_N)_i^n & (\mathbf{y}_N)_i^n & \cdots & 1 \end{bmatrix}$,

$\mathbf{a} = \begin{bmatrix} a_n & a_{n-1} & \cdots & a_0 \end{bmatrix}^T$, and $\mathbf{b} = \begin{bmatrix} (\mathbf{x}_1)_i & \cdots & (\mathbf{x}_N)_i \end{bmatrix}^T$,

then the coefficient vector $\mathbf{a}$ of the polynomial that minimizes J_i is just the least-square solution,

$$\hat{\mathbf{a}} = \left(\mathbf{Y}^T \mathbf{Y} \right)^{-1} \mathbf{Y}^T \mathbf{b} . \quad (12)$$

Practically, the order n of the polynomial $\mathcal{T}_i(.)$ cannot be too large to prevent from the ill-conditioned matrix $\mathbf{Y}^T\mathbf{Y}$. When $n = 1$, the transformation $\mathcal{T}_i(.)$ is a linear function, and is often called linear regression (LR) or linear least square regression. Similarly, when $n = 2$, $\mathcal{T}_i(.)$ is a quadratic function and is called quadratic least square (QLS) regression.

In particular, for the linear regression case ($n = 1$), with eq. (12) we obtain the transform function as

$$\mathcal{T}_i\left((\mathbf{y})_i \right) = \rho \frac{\sigma_{\mathbf{x},i}}{\sigma_{\mathbf{y},i}} \times \left[(\mathbf{y})_i - \mu_{\mathbf{y},i} \right] + \mu_{\mathbf{x},i} , \quad (13)$$

where

$$\rho = \frac{1}{N} \left(\sum_{m=1}^{N} \left((\mathbf{x}_m)_i - \mu_{\mathbf{x},i} \right) \left((\mathbf{y}_m)_i - \mu_{\mathbf{y},i} \right) \right) \Big/ \left(\sigma_{\mathbf{x},i} \sigma_{\mathbf{y},i} \right) , \quad (14)$$

which is called the correlation coefficient.

Comparing eq. (13) with eq. (6), we find LR and CSC are quite similar. The only difference is that LR considers the correlation between $\mathbf{y}_m$ and $\mathbf{x}_m$ while CSC simply assumes they are completely associated ($\rho = 1$). Note that in fact $\rho < 1$ because $\mathbf{y}_m$ is not a linear function of $\mathbf{x}_m$, which can be shown by combining equations (1)-(3):

$$\mathbf{y}_m = \mathbf{C} \log\left(\exp\left(\mathbf{C}^{-1} \mathbf{x}_m \right) + \tilde{\mathbf{n}} \right) . \quad (15)$$

4. Experimental Setup

The proposed codebook-based algorithms have been tested with the AURORA2 database. For the recognition experiments, two sets (Sets A and B) of utterances artificially contaminated by different types of noise (subway, babble, car, etc.) and different SNR levels (ranging from -5dB to 20dB) were prepared. Since the proposed algorithms only involve the front-end feature extraction, all the procedures for training and recognition are identical to the reference experiments stated in the AURORA2 documentation [3].

For the clean training database, each of the 8440 strings was first converted into a stream of 23 mel-spectral coefficients plus log-energy. All of these 24-dimensional feature vectors were used to construct a set of *N* codewords via vector quantization (VQ). These codewords were also converted to 13-dimensional cepstral vectors (c0~c12 and log-energy) to form the clean speech cepstral codebook $\{\mathbf{x}_m\}$. The size *N* of the codebook is set to give the best recognition performance. In addition, all of these 24-dimensional feature vectors in the training set were converted to cepstral domain. The obtained MFCC features plus their delta and delta-delta were the components in the finally used 39-dimensional feature vectors. With these feature vectors in the training database the hidden Markov models for each digit were trained.

For the testing condition, we estimated the noise vector for each utterance by simply averaging its first 5 mel-spectral frames. That is, we assumed the first 5 frames (about 65 ms) of each utterance contain noise only. Then, following the procedures stated in section 2, we constructed the noise corrupted speech cepstral codebook $\{\mathbf{y}_m\}$. Then based on the two codebooks $\{\mathbf{x}_m\}$ and $\{\mathbf{y}_m\}$, the various proposed algorithms were performed to adjust the testing features, respectively.

5. Experimental Results and Discussions

Table 1 lists the recognition results of MFCC baseline and several robustness approaches, including utterance-based CMN (U-CMN) and CMVN (U-CMVN), codebook–based CMN (C-CMN) and CMVN (C-CMVN), linear regression (LR), quadratic least square (QLS) regression and two versions of CSC, where CSC-1 compensates only mean values as in eq. (5) and CSC-2 compensates both mean and variance values as in eq. (6). From this table, several phenomena can be observed:

1. When the training and testing are in matched clean condition, all approaches give very similar high recognition accuracy, which means these robustness techniques do not reduce the discriminating capability of MFCC.
2. For the conventional utterance-based CMN and CMVN, the performance improvements are obvious, and CMVN is especially better than CMN for the test A set under lower SNR conditions (0~10dB). However, for the test B set, their corresponding recognition rates are quite similar.
3. Both of the two CSC approaches enhance the noise robustness of the original MFCC features significantly. For example, CSC-1 gives about 7% and 12% word accuracy improvements for sets A and B, respectively, and CSC-2 gives 13% and 17% for sets A and B, respectively. As a result, CSC-2 is obviously better than CSC-1, which implies further compensating the variance is very helpful. Furthermore, when compared with utterance-based CMN and CMVN, CSC-2 is apparently superior to CMVN for all noise conditions, while CSC-1 outperforms CMN only for test set A.

Test	System	clean	20dB	15dB	10dB	5dB	0dB	-5dB	Average (0~20dB)
Test Set A	Baseline	98.91	94.99	86.93	67.28	39.36	17.07	8.40	61.13
	U-CMN	98.98	96.90	92.30	76.15	43.37	22.05	12.85	66.15
	U-CMVN	98.98	95.98	91.66	80.48	57.40	26.40	10.96	70.38
	CSC-1	98.94	96.79	92.69	79.20	50.90	21.60	10.55	68.24
	CSC-2	98.94	97.11	94.68	86.35	63.96	28.86	10.15	74.20
	C-CMN	98.94	96.79	92.69	79.20	50.90	21.60	10.55	68.24
	C-CMVN	98.95	97.21	94.48	86.01	64.35	30.28	10.90	74.47
	LR	98.91	97.20	94.71	87.35	66.70	32.99	12.04	75.79
	QLS	98.94	97.16	94.86	87.28	67.51	33.65	11.61	76.09
Test Set B	Baseline	98.94	92.35	80.79	58.06	32.04	14.63	7.92	55.57
	U-CMN	98.98	97.63	94.15	82.19	52.34	26.12	14.05	70.49
	U-CMVN	98.98	96.41	92.15	81.78	58.69	26.47	10.98	71.10
	CSC-1	98.94	96.61	91.73	77.19	50.82	22.87	10.19	67.84
	CSC-2	98.94	97.19	94.03	84.55	60.25	27.78	10.82	72.76
	C-CMN	98.94	96.61	91.73	77.19	50.82	22.87	10.19	67.84
	C-CMVN	98.95	97.02	93.63	83.56	59.59	28.31	11.05	72.42
	LR	98.91	97.30	94.65	86.29	64.64	31.50	12.00	74.88
	QLS	98.94	97.00	93.27	84.00	63.43	32.43	12.07	74.02

Table 1. Recognition accuracy (%) for baseline and various approaches, utterance-based CMN (U-CMN), utterance-based CMVN (U-CMVN), mean compensated CSC (CSC-1), mean-and-variance compensated CSC (CSC-2), codebook-based CMN (C-CMN), codebook-based CMVN (C-CMVN), linear regression (LR) and quadratic least square (QLS), on Test Sets A and B of Aurora 2 database.

4. Comparing the different types of CMN and CMVN, it shows that the codeword-based CMN and CMVN are better than the utterance-based ones, respectively, in set A conditions. (In fact, it can be proved that codebook-based CMN is equivalent to CSC-1). Thus we may roughly conclude that the statistics estimated from the codebooks are more accurate than those from per utterance especially for stationary noise environments like set A.
5. Finally, for the two regression approaches, we find both LR and QLS also significantly improves the recognition accuracy. In fact, LR and QLS are always better than the other all approaches. LR outperforms CSC-2 probably because it further considers the correlation between clean and noisy conditions, as stated in section 3. Also, QLS is originally expected to be better than LR since it uses higher-order polynomial and statistics, but it is not always the case. This is probably because the matrix $\mathbf{Y}^T\mathbf{Y}$ in eq. (9) is nearly ill-conditioned, or the over-fitting problem happens in QLS.

6. Comparison with some other existing algorithms

In the former researches, there are a series of approaches based on VQ codebooks for training and/or testing environments, like CDCN [4] and FCDCN [5]. Since no Expectation Maximization (EM) techniques or codeword-selection procedures are needed, the proposed codebook–based algorithms in this paper differ from these approaches primarily in their simplicity for realization. In addition, similar to feature-space MLLR [6], most of our proposed codebook-based approaches transform the testing features linearly. However, feature-space MLLR and all of them utilize different optimization criteria, and again they seem easier to perform than feature-space MLLR. According to [7], CDCN gives 71.74% and 73.36% and feature-space MLLR gives 72.27% and 74.72% averaged recognition rates for Aurora-2 Test sets A and B, respectively. Therefore, even if the proposed algorithms are simpler for realization, they can perform as well as, or sometimes even better than, the CDCN and feature-space MLLR.

7. Conclusions and Future Works

In this paper, we design pseudo stereo codebooks that respectively represent the clean and noisy conditions. With these codebooks some novel noise robustness algorithms are proposed, including cesptral statistics compensation (CSC), codebook-based CMN and CMVN, linear regression (LR) and quadratic least square (QLS) regression methods. We show that all of them effectively improve the recognizer's performance in additive noise environments. It is expected that these codebook-based approaches can be further enhanced if a voice activity detector (VAD) is applied to obtain the accurate noise estimate. Furthermore, we believe that the well known Histogram Equalization (HEQ) that roughly normalizes all statistics can be realized based on pseudo stereo codebooks, which can further improve the noise robustness of speech features.

References

[1] O. Viikki, K. Laurila, "Cepstral domain segmental feature vector normalization for noise robust speech recognition," Speech Communication, 1998

[2] S. Furui, "Cepstral analysis technique for automatic speaker verfication," IEEE Trans. on Acoustics, Speech Signal Processing, 1981

[3] H.-G Hirsch, D. Pearce, "The AURORA experimental framework for the performance evaluation of speech recognition systems under noisy conditions," ISCA ITRW ASR 2000, Paris, France, September 18-20, 2000

[4] A. Acero, "Acoustical and environmental robustness in automatic speech recognition," Ph.D.dissertation, Dept. Elect. Comput. Eng., Carnegie Mellon University, Pittsburgh, PA, 1990.

[5] A. Acero and Richard M. Stern, "Environmental robustness in automatic speech recognition", ICASSP 1990

[6] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," Eng. Dept., Cambridge Univ., Cambridge, U.K., CUED/F-INFENG, 1997.

[7] J. M. Huerta, "Alignment-based codeword-dependent Cepstral Normalization" IEEE Trans. on Speech and Audio Processing, October 2002