Modified Cepstral Mean Normalization - Transforming to utterance specific non-zero mean

Vikas Joshi[1,2],*N. Vishnu Prasad*[1], *S. Umesh*[1]

[1] Department of Electrical Engineering, Indian Institute of Technology, Madras
[2] IBM India Research Labs, Bangalore

vijoshi7@in.ibm.com, {ee12s021,umeshs}@ee.iitm.ac.in

Abstract

Cepstral Mean Normalization (CMN) is a widely used technique for channel compensation and for noise robustness. CMN compensates for noise by transforming both train and test utterances to zero mean, thus matching first-order moment of train and test conditions. Since all utterances are normalized to zero mean, CMN could lead to loss of discriminative speech information, especially for short utterances. In this paper, we modify CMN to reduce this loss by transforming every noisy test utterance to the estimate of clean utterance mean (mean estimate of the given utterance if noise was not present) and not to zero mean. A look-up table based approach is proposed to estimate the clean-mean of the noisy utterance. The proposed method is particularly relevant for IVR-based applications, where the utterances are usually short and noisy. In such cases, techniques like Histogram Equalization (HEQ) do not perform well and a simple approach like CMN leads to loss of discrimination. We obtain a 12% relative improvement over CMN in WER for Aurora-2 database; and when we analyze only short utterances, we obtain a relative improvement of 5% and 25% in WER over CMN and HEQ respectively.

Index Terms: Robust speech recognition, CMN, CMVN, HEQ

1. Introduction

The performance of a speech recognition system degrades under noisy environments due to mismatch between train and test condition. Numerous approaches have been proposed for noise compensation for robust speech recognition [1, 2, 3, 4, 5, 6]. Addition of noise changes the statistics of the clean signal including the mean, variance and other higher order moments. The simplest and most widely used technique for noise compensation is Cepstral Mean Normalization (CMN) [3, 7] which compensates for the effect of the noise on the mean of clean distribution. Similarly, Cepstral Mean and Variance Normalization (CMVN) [2] transforms every noisy utterance, such that mean and variance of transformed utterance match with the global mean and the variance of clean data. Histogram Equalization (HEQ) [8, 9, 5, 10] is an extension to CMVN where the entire histogram (i.e. all moments) of every noisy utterance is matched to clean speech histogram.

In many Interactive Voice Response (IVR) systems, the user query will typically have short utterances (one or two words spoken) as the input. Building Automatic Speech Recognition (ASR) could still be challenging since it has to recognize these short utterances under noisy conditions. Noise compensation techniques like HEQ may not be very suitable for short utterances. The performance of HEQ degrades for short utterances due to a) less data available to estimate the utterance histogram b) loss of discriminative speech information, since every short utterance is forced to match a same clean histogram. VTS is shown to perform well even for short utterances [6], but the computational complexity of VTS is high [11], making it unsuitable for applications that require real time response. Simple approaches like CMN works well in case of short utterances and hence improvement over CMN could still be important.

1.1. Motivation

CMN was introduced to compensate for convolutive noise [3, 7]. In case of additive noise, CMN compensates for the effect of noise on mean of clean speech distribution. Consider a clean cepstral vector $\mathbf{x}$ (with 13 dimensions) contaminated with noise $\mathbf{n}$ (additive in cepstral domain) to obtain noisy cepstra $\mathbf{y}$. Contamination by noise would result in shift of clean cepstral mean from $\boldsymbol{\mu}_x$ to $\boldsymbol{\mu}_y$ for every component i in feature vector, as in Eqn. (1).

$$y^i = x^i + n^i; \qquad \mu_y^i = \mu_x^i + \mu_n^i \quad i = 0, 1, 2.., 12 \quad (1)$$

where μ_n^i is the mean of noise alone for i^{th} component. In CMN, both train and test utterances are subtracted from its mean as follows:

$$\hat{x}^i = x^i - \mu_x^i; \quad \hat{y}^i = y^i - \mu_y^i \qquad \Longrightarrow \qquad \mu_{\hat{y}}^i = \mu_{\hat{x}}^i = 0 \quad (2)$$

Thus, after normalization, mean of every transformed train and test utterance (i.e., $\mu_{\hat{x}}^i$ and $\mu_{\hat{y}}^i$) is equal (zero) as shown by Eqn. (2), thus compensating the effect of the noise on the mean of the clean speech distribution. This is done separately for every component in feature vector.

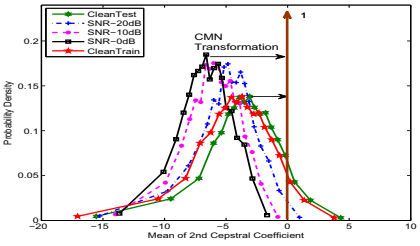


Figure 1: ***Histogram of mean*** of utterances for 2^{nd} cepstral coefficient under different noise conditions for Aurora-2 test data-set. Figure also shows the effect of CMN transformation on the histogram of utterance means

In practice, every feature component of the utterance has a distinct mean value and hence *component means* will inturn have a distribution (with certain variance). From every utter-

ance one single mean value is obtained. Histogram is then plotted using all the mean values obtained from all the utterances. Fig. 1 shows the *histogram of mean* for 2^{nd} cepstral coefficient for all the utterances under different noise conditions, for Aurora-2 test data-set. Note that the plot in Fig. 1, is histogram of *utterance mean* for 2^{nd} cepstral coefficient and not histogram of 2^{nd} cepstral coefficient itself (which is used in HEQ). Since CMN transforms all the utterances to zero mean under both train and test conditions, the probability density function (pdf) of the utterance mean after CMN, is a delta function at zero. Hence *CMN does not preserve the shape of mean distribution*, which essentially corresponds to loss of some useful discriminative information between sound classes.

In this paper, we attempt to eliminate the disadvantage of CMN by reducing the loss in speech information. If we could transform every cepstra of the noisy utterance (with mean μ_y) to its corresponding clean utterance mean ($\boldsymbol{\mu}_x$), then no useful information is lost and yet we can still compensate the effect of noise. This is shown in the Eqn. (3).

$$\hat{\mathbf{y}} = \mathbf{y} - \boldsymbol{\mu}_y + \boldsymbol{\mu}_x \Rightarrow \boldsymbol{\mu}_{\hat{y}} = \boldsymbol{\mu}_x \tag{3}$$

If this is feasible, then all utterances are transformed to its clean mean and not to a single common mean (zero mean) as is normally done in CMN. An oracle experiment using the stereo data in Aurora-2 database was conducted to validate the above hypothesis. Each noisy utterance was transformed to its clean mean using Eqn. (3). The clean mean of the noisy utterance was obtained from clean version of the noisy utterance and hence we call it an oracle experiment. The result obtained (refer to Tables 1 and 2) indicates that, transforming to utterance specific mean increases the recognition accuracy compared to CMN.

However, in practice, given a noisy test utterance, its corresponding clean utterance mean ($\boldsymbol{\mu}_x$) is not known. We propose a look-up table based approach to get an *estimate of the clean utterance mean* from the given noisy utterance. Then, the estimate of clean utterance mean ($\hat{\boldsymbol{\mu}}_x$) is used in Eqn. (3) instead of $\boldsymbol{\mu}_x$. If the estimate $\hat{\boldsymbol{\mu}}_x$ is close to the true mean $\boldsymbol{\mu}_x$, then the loss in the speech information can be reduced while compensating for noise. Creating the look-up from the training data and algorithm to estimate the clean utterance mean are discussed in detail in section 2. We use the term Utterance Specific Mean Normalization (USMN) to refer to the approach of transforming an utterance to its clean mean.

Our analysis show that USMN preserves the distribution of the *utterance means* even after normalization, while CMN does not. USMN shows a 12% relative improvement over CMN in WER for Aurora-2 database; and when analyzed with only short utterances, USMN has a relative improvement of 5% and 25% in WER over CMN and HEQ respectively.

The rest of the paper is organized as follows. In section 2 USMN approach is explained in detail followed by analysis of USMN approach in section 3. Section 4 contains discussion on experimental setup and followed by recognition results in section 5 . Finally conclusions are presented in section 6.

2. Utterance Specific Mean Normalization

In USMN, every noisy cepstra is normalized using the estimate of corresponding clean utterance mean as shown below,

$$\hat{\mathbf{x}} = \mathbf{y} - \boldsymbol{\mu}_y + \hat{\boldsymbol{\mu}}_x \tag{4}$$

where $\mathbf{y}$ is the 13 dimensional noisy cepstra, $\boldsymbol{\mu}_y$ is the mean of noisy cepstra $\mathbf{y}$, $\hat{\boldsymbol{\mu}}_x$ is the estimate of the clean mean of $\mathbf{y}$ and $\hat{\mathbf{x}}$ is the transformed cepstra. However, for a given noisy utterance, its corresponding clean mean $\hat{\boldsymbol{\mu}}_x$ is not known. The algorithm to estimate the clean utterance mean from the given noisy utterance is explained next.

2.1. Algorithm - Estimation of clean utterance mean

To estimate the clean utterance mean of the noisy signal, we use the mathematical model that describes the effect of noise (additive or convolutive). In this paper we discuss the approach to estimate clean mean for case of additive noise alone and convolutive noise alone. Mixture of convolutive and additive noise is not addressed in this paper.

2.1.1. USMN for Additive Noise

Let y_t be the observed noisy speech, x_t is the clean speech and n_t is the additive noise. Then the effect of additive noise in the time domain is given by,

$$y_t = x_t + n_t$$

Then, finding the magnitude square Fourier Transform, we get

$$|y(\omega)|^2 = (x(\omega) + n(\omega))(x^*(\omega) + n^*(\omega))$$

where $y(\omega)$, $x(\omega)$ and $n(\omega)$ are Fourier transform of noisy speech, clean speech and additive noise. Assuming speech and noise to be uncorrelated and applying log compression, we get,

$$log(Y(\omega)) = log(X(\omega)) + log(1 + N(\omega)/X(\omega))$$

where $Y(\omega)$, $X(\omega)$ and $N(\omega)$ be the squared magnitude Fourier coefficients of corrupted speech, clean speech and additive noise (e.g. $Y(\omega) = |y(\omega)|^2$). Applying DCT transformation, $\mathbf{D}$, we get,

$$\mathbf{y} = \mathbf{x} + \mathbf{D} * log(\underline{\mathbf{1}} + e^{\mathbf{D}^{-1}(\mathbf{n}-\mathbf{x})}) \tag{5}$$

where $\mathbf{y}$, $\mathbf{x}$ and $\mathbf{n}$ are 13 dimensional feature vectors of corrupted noisy cepstra, clean cepstra and noise cepstra respectively. $\mathbf{D}$ is the DCT transformation Matrix. Taking the expectation (denoted by $\mathbb{E}$) of Eqn. (5), we get,

$$\boldsymbol{\mu}_x = \boldsymbol{\mu}_y - \mathbb{E}(\mathbf{D}[log(\underline{\mathbf{1}} + e^{\mathbf{D}^{-1}(\mathbf{n}-\mathbf{x})})])$$

$$\boldsymbol{\mu}_x = \boldsymbol{\mu}_y - v(\mathbf{n}, \mathbf{x}) \tag{6}$$

The goal is to find $\boldsymbol{\mu}_x$ from Eqn. (6). Calculating $v(\mathbf{n}, \mathbf{x})$ (i.e., expectation over random variables $\mathbf{n}$ and $\mathbf{x}$) is difficult, since probability distribution of $log(\underline{\mathbf{1}} + e^{-\mathbf{D}^{-1}(\mathbf{n}-\mathbf{x})})$ is not known, even for the case when both $\mathbf{n}$ and $\mathbf{x}$ are assumed Gaussian. Approximating $\mathbf{D}[log(\underline{\mathbf{1}} + e^{\mathbf{D}^{-1}(\mathbf{n}-\mathbf{x})})$ by the zero-order vector taylor series around noise mean ($\boldsymbol{\mu}_n$) and clean speech mean ($\boldsymbol{\mu}_x$) as done in VTS [6] approach, we get,

$$v(\mathbf{n}, \mathbf{x}) \simeq \mathbf{D}[log(\underline{\mathbf{1}} + e^{\mathbf{D}^{-1}(\boldsymbol{\mu}_n - \boldsymbol{\mu}_x)})]$$

Substituting for $v(\mathbf{n}, \mathbf{x})$ in Eqn. (6) and rearranging we get,

$$\boldsymbol{\mu}_x - \boldsymbol{\mu}_y + \mathbf{D}[log(\underline{\mathbf{1}} + e^{\mathbf{D}^{-1}(\boldsymbol{\mu}_n - \boldsymbol{\mu}_x)})] = \mathbf{0} \tag{7}$$

In our experiments, $\boldsymbol{\mu}_y$ is obtained as the mean of the entire utterance and $\boldsymbol{\mu}_n$ is obtained as mean of silence (noise alone) frames. Similar to VTS, we assume first twenty and last twenty frames contain only noise and no speech. Note that, unlike VTS we are interested only in *mean* of the clean utterance and not in obtaining $\mathbf{x}$ itself.

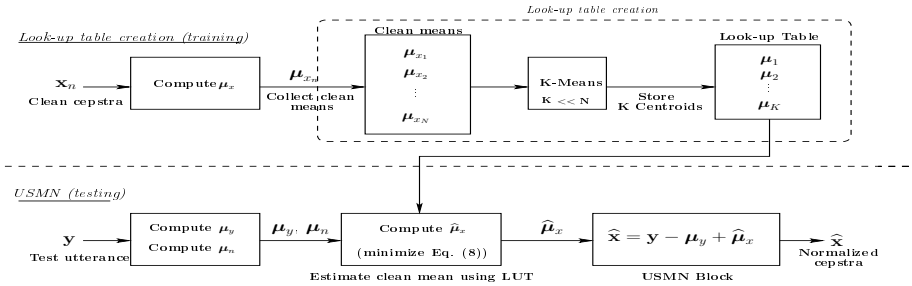


Figure 2: Block diagram to create Look-up table (LUT) and steps in performing USMN normalization of noisy test utterances

Even with the knowledge of $\boldsymbol{\mu}_y$ and $\boldsymbol{\mu}_n$ it is not possible to obtain a closed form solution for $\boldsymbol{\mu}_x$ from the Eqn. (7). Alternatively, we can search for $\boldsymbol{\mu}_x$ in 13 dimensional space to minimize the $l_2 - norm$ of the below Eqn. (8).

$$e_v = [\hat{\boldsymbol{\mu}}_x - \boldsymbol{\mu}_y + \mathbf{D}[log(\underline{\mathbf{1}} + e^{\mathbf{D}^{-1}(\boldsymbol{\mu}_n - \hat{\boldsymbol{\mu}}_x)})]]$$
$$e_v^T e_v = e \tag{8}$$

This search in unconstrained 13 dimensional space is computationally very expensive. Hence we use a look-up table (LUT) based approach where μ_x is chosen from a set of mean values, which minimizes the error e in Eq. (8). LUT is created using mean values of training utterances as shown in Fig. 2. Here we assume that the mean of the test utterances are similar to the mean of train utterances, i.e. training data contains most of clean utterance means that can occur during testing. Thus for a given noisy utterance, the estimate of clean mean is obtained by choosing the nearest mean from the training utterances itself. Furthermore, size of LUT is reduced (for computational benefits) by clustering the means using K-means algorithm and choosing the K cluster centroids as representative mean vectors. Finally, LUT has K number of 13 dimensional mean vectors as shown in Fig. 2. This trade-off between the computational gain to loss in perforamnce is discussed in section 5.

We next discuss the steps to obtain utterance specific mean normalized features ($\hat{x}$) from given noisy test utterances (y) and is shown in the Fig. 2. The 3 step process is discussed below:

- Firstly, the noisy utterance mean, μ_y, and noise mean μ_n are computed. μ_y is computed as sample mean of all the frames in the utterance and μ_n is computed as sample mean of first and last twenty frames.
- $\hat{\mu}_x$ is then estimated by choosing one of K-values from the LUT, which minimizes error e from Eqn. (8).
- Finally, noisy utterance is normalized using the estimated $\hat{\mu}_x$ according to Eqn. (9).

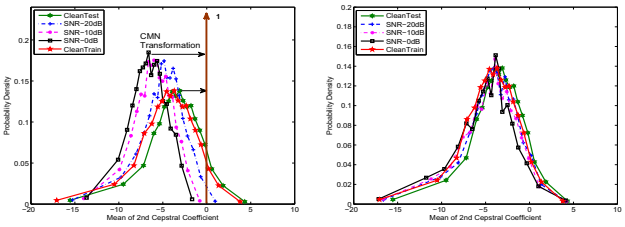
$$\hat{x} = y - \mu_y + \hat{\mu}_x \tag{9}$$

2.1.2. USMN for Convolutive Noise

Convolutive noise (h_t) in the time domain becomes an additive term in the cepstral domain (h).

$$y_t = x_t * h_t; \qquad y = x + h \Rightarrow \mu_y = \mu_x + \mu_h$$
$$\mu_x - \mu_y + \mu_h = 0 \tag{10}$$

Here the estimate of the clean mean can be easily obtained according to Eqn. (10). μ_y can be approximated by the sample mean of the utterance as done in the additive noise case. μ_h can be approximated by sample mean of the silence frames. Thus an estimate of the clean utterance mean can be obtained using Eqn. (10) and with the knowledge of μ_y and μ_h.



a) Mean Histogram - with and without CMN b) Estimated Mean Histogram (USMN)

Figure 3: Histograms of *mean of utterances* for 2^{nd} cepstral coefficient for Aurora-2 database for a) with and without CMN b) USMN approach. Histogram of means show a better match between clean and noisy condition after performing USMN.

2.1.3. Training and testing phase

In USMN approach, normalization is done only on test features and *not* on train features (unlike CMN, HEQ or VTS). During training phase, HMM models are directly built from standard MFCC features. During testing, noisy features are first normalized with their estimated clean mean as show in the Fig. 2 and are then used for recognition.

3. Analysis

In this section we analyze the efficacy of using the proposed approach to estimate mean of the corresponding clean utterance from given noisy utterance. We study the statistical behavior of the estimated means by comparing the distribution of estimated means under different noisy conditions. Fig. 3(a) shows the *histogram of mean of utterances* for 2^{nd} cepstral coefficient, for clean train utterances, clean test utterances and for utterances under different SNR conditions of Aurora-2 database. It can be seen that noise distorts the mean distribution. Fig. 3(b) show the *histogram of estimated clean means of noisy utterance using proposed approach* for 2^{nd} cepstral coefficient under different noise conditions for same data-set. Comparing Fig. 3(a) and 3(b), following observations can be made.

- Histograms of estimated means under noisy conditions closely match histogram of clean means. Hence estimation of means is accurate enough with proposed approach. This is also asserted by the improvement in the recognition results over CMN (Table 1, Table 2).
- In USMN, shape of mean distribution of train utterances is preserved. Preserving the distribution of mean would correspond to retaining the individual utterance mean values and thus preserving the speech information as discussed in section 1.1. In contrast CMN maps all the means to zero, effectively making variance of mean distribution to zero.

4. Experimental Setup

Database: We test the performance of USMN on Aurora-2 database, comprising of connected spoken digits contaminated with different types of noise at various SNR levels [12]. Since CMN is preferred for applications having short utterances, we compare the performance of USMN approach, CMN and HEQ for complete test data-set and also for short utterances separately. Utterances having a maximum of two spoken digits are considered as short utterances. Entire test data set inclusive of all noise conditions have 70070 utterances, out of which 29799 are short utterances (having one or two spoken digits).

Feature Extraction and Acoustic Modeling: HMM Toolkit (HTK) 3.4 is used for experiments. Standard MFCC vectors are used for basic feature parametrization. Short time Fourier transform of pre-emphasized speech signal is obtained using $25ms$ window and shift size of $10ms$. 23 mel-scaled filter banks are used for smoothing the spectrum. 13 dimensional cepstral coefficients are used (inclusive of C_0). Utterance-wise subtraction of the mean value of each cepstral coefficient is done to compute CMN features. HEQ features are obtained by transforming the utterances to match clean speech CDF as done in [8]. Clean speech CDF is obtained from all the train utterances. In oracle experiment, each noisy test speech file is normalized to its own clean mean, since its clean version is available from the database. Finally 13 delta and 13 acceleration coefficients are appended to get composite 39 dimensional MFCC vector per frame. The acoustic model is a left to right continuous density HMM with 16 states and 3 diagonal covariance Gaussian mixtures per state. Word level HMM model is used. Training is done using clean train utterances from Aurora-2 data-set.

5. Results & Discussion

We study the performance of CMN, HEQ and USMN on both long and short utterances. Table 1 compares the performance for all utterances (both long and short) and Table 2 records the accuracies of short utterances only.

Table 1: Recognition results - Aurora-2

	Baseline	CMN	USMN (Oracle)	USMN	HEQ
Clean	99.12	99.2	99.18	99.12	99.07
SNR20	95.49	97.35	97.45	97.20	97.57
SNR15	84.85	93.43	93.88	93.49	95.38
SNR10	60.39	80.62	82.25	82.29	89.73
SNR5	30.70	51.87	59.05	58.99	75.26
SNR0	13.24	24.30	34.09	33.17	44.63
SNR-5	8.15	12.03	19.40	16.62	16.33
Average	56.93	69.51	73.35	**73.03**	80.51

Table 2: Recognition results - Aurora-2 SHORT utterances

	Baseline	CMN	USMN (Orcale)	USMN	HEQ
Clean	99.47	99.49	99.43	99.45	98.87
SNR20	92.61	98.22	98.27	97.97	95.84
SNR15	72.98	95.91	95.73	95.95	92.55
SNR10	32.77	88.54	87.41	88.54	84.19
SNR5	-5.59	68.60	69.49	70.06	68.49
SNR0	-10.29	43.77	51.32	47.82	38.68
SNR-5	-2.58	22.12	34.41	26.04	14.65
Average	36.49	79.01	**80.44**	**80.07**	75.95

For long utterances, USMN consistently outperforms CMN. However HEQ is better than CMN and USMN. Oracle experiment preforms better than CMN under for all noise types and thus asserting the need for utterance specific mean normalization. Oracle experiment shows that normalization with utterance speicfic mean becomes more important as SNR degrades. Performance of USMN closely matches oracle experiment thereby asserting the appropriateness of using the proposed approach to estimate the clean utterance mean. The performance of HEQ degrades significantly in case of short utterances. CMN, performs better than HEQ for short utterances & USMN has higher overall accuracy in comparison with both CMN and HEQ.

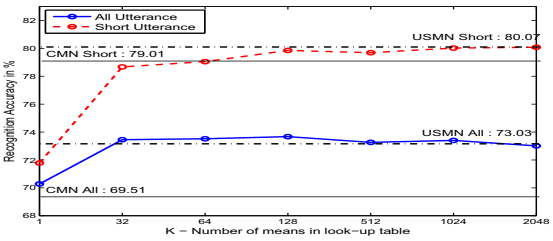


Figure 4: Recognition accuracy for different look-up table size

We also study the trade-off between the performance and computational gain by reducing the size of look-up table used in USMN. Fig. 4 shows the plot of recognition accuracy as the number of clusters (K) is varied from 1 to 2048. Cluster size of 1 would represent a single mean (global mean of all train utterances). As the number of clusters are increased, the performance improves and is seen to plateau after 128 cluster points. Average time to normalize a short utterance (as run on our Intel Core 2 Duo laptop) with 2048 clusters is $\sim 15ms$ and was seen to reduce by $\sim 5x$ times for 128 clusters and thus compensation is real time. The above advantages of USMN increases its relevance in context of real-time IVR systems.

6. Conclusions

In this paper we have presented a feature normalization technique, USMN, that can reduce the loss of speech information during CMN. Some of discriminative speech information is lost in CMN approach by normalizing each utterance to zero mean. We attempt to overcome this particular disadvantage of CMN by normalizing each utterance to its clean mean. A look-up table-based approach to estimate the clean mean from the given noisy utterance was proposed. Analysis show that the histogram of *estimated mean values* under different noisy conditions match closely with actual mean histograms. Recognition results show improvements over CMN for both long and short utterances. USMN approach is well-suited for IVR kind of applications which have short amount of data and need quick response time.

7. Acknowledgments

This work was supported under the SERC project funding SR/S3/EECE/058/2008 of Department of Science and Technology, India. This work is part of Vikas's work towards PhD at IIT Madras. Vikas would like to thank IBM for the support.

8. References

[1] R. Balchandran and R. Mammone, "Non-parametric estimation and correction of non-linear distortion in speech system," in *ICASSP*, 1998.

[2] O. Viikki and K. Laurila, "Cepstral domain segmental feature vector normalization for noise robust speech recognition," *Speech Communications*, 1998.

[3] S. Furui, "Cepstral analysis technique for automatic speaker verification," *IEEE Transaction on Acoustics, Speech and Signal Processing*, vol. 29, pp. 254–272, 1981.

[4] Y. Gong, "Speech recognition in noisy environments: A survey," CRIN/ CNRS - INRIA-Lorraine, Nancy, France, Tech. Rep., Nov. 1994.

[5] F. Hilger and H. Ney, "Quantile based histogram equalization for noise robust large vocabulary speech recognition," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 14, no. 3, pp. 845 – 854, may 2006.

[6] P. J. Moreno, B. Raj, and R. M. Stern, "A vector taylor series approach for environment-independent speech recognition," in *Proc. ICASSP-96*, 1996, pp. 733–736.

[7] O. M. Strand and A. Egeberg, "Cepstral mean and variance normalization in the model domain," in *ISCA Tutorial and Research Workshop*, 2004.

[8] A. de la Torre, A. Peinado, J. Segura, J. Perez-Cordoba, M. Benitez, and A. Rubio, "Histogram equalization of speech representation for robust speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 3, pp. 355 – 366, May 2005.

[9] S. Molau, M. Pitz, and H. Ney, "Histogram based normalization in the acoustic feature space," in *ASRU*, 2001.

[10] F. Hilger, S. Molau, and H. Ney, "Quantile based histogram equalization for online applications," in *Interspeech*, 2002.

[11] Y. Obuchi and R. Stern, "Normalization of time-derivative parameters using histogram equalization," in *Proc. of EUROSPEECH 2003, Geneva, Switzerland*, 2003.

[12] D. Pearce and H.-G. Hirsch, "The aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *in ISCA ITRW ASR2000*, 2000, pp. 29–32.