論文 / 著書情報
Article / Book Information

Title	Target Speech GMM-based Spectral Compensation for Noise Robust Speech Recognition
Authors	Takahiro Shinozaki, Sadaoki Furui
Citation	INTERSPEECH 2009 BRIGHTON, , , pp. 1255-1258,
Pub. date	2009, 9
Copyright	(c) 2009 International Speech Communication Association, ISCA
DOI	http://dx.doi.org/

INTERSPEECH 2009 BRIGHTON

Target Speech GMM-based Spectral Compensation for Noise Robust Speech Recognition

Takahiro Shinozaki, Sadaoki Furui

Department of Computer Science, Tokyo Institute of Technology, Tokyo, Japan
www.furui.cs.titech.ac.jp

Abstract

To improve speech recognition performance in adverse conditions, a noise compensation method is proposed that applies a transformation in the spectral domain whose parameters are optimized based on likelihood of speech GMM modeled on the feature domain. The idea is that additive and convolutional noises have mathematically simple expression in the spectral domain while speech characteristics are better modeled in the feature domain such as MFCC. The proposed method works as a feature extraction front-end that is independent from decoding engine, and has ability to compensate for non-stationary additive and convolutional noises with a short time delay. It includes spectral subtraction as a special case when no parameter optimization is performed. Experiments were performed using the AURORA-2J database. It has been shown that significantly higher recognition performance is obtained by the proposed method than spectral subtraction.

Index Terms: noisy speech recognition, spectrum, Gaussian mixture model

1. Introduction

Speech recognition performance in real environment is largely affected by additive and convolutional noises. The additive noise is associated with sound waveforms other than target speech to recognize and the convolutional noise corresponds to channel characteristics. To achieve high recognition performance, compensations for both of these noises are very important. In the spectral domain, additive and convolutional noises are simply expressed as additive and multiplicative terms to original speech spectrum. However, during the feature extraction process for speech recognition such as MFCC [1], the noise effects are diffused across spectral axis and are wound due to the filter bank analysis, log transform, etc and the relation between noise and speech become complicated. Therefore, the most direct noise compensation approach would be applying a reverse affine transformation in the spectral domain before filter banking.

Spectral subtraction [2] is one of the most popular noise robustness techniques that works by subtracting a noise vector from noisy speech in the spectral domain. The noise vector is estimated from a non-speech segment. While it is very effective for additive noise, a limitation is that it does not compensate for convolutional noise. Since it assumes the noise vector is constant, it also lacks an ability to follow the changes of additive noise. Feature-space stochastic matching [3] applies an affine transformation in the spectral domain so that features obtained from the transformed spectrum match better with HMM used for speech recognition. In the study by Sankar et. al. [3], HMM likelihood was used as the objective function where the state alignment was obtained using a recognition hypothesis. While the framework is designed to compensate both for additive and convolutional noises, the experiment was limited to additive noise in their study. Experiments that compensate both types of noises were performed by Kim et. al. [4] in which the transformation coefficients were estimated offline based on likelihood that was computed with manually transcribed labels. A disadvantage of these approaches is that the framework uses HMM likelihood. Since this requires state alignment, it is not suitable for online decoding.

In this paper, we propose a spectral compensation method that applies a transformation in the spectral domain before filter banking, in which parameters of the transformation are optimized based on speech GMM likelihood. Since GMM is used instead of HMM, the proposed method works as a feature extraction front-end that is independent from decoder without requiring state alignment. Although GMM can be regarded as a special case of HMM, our parameter estimation algorithm is different from the previous studies in that it introduces a continuous function approximation and has an advantage that a flooring operation necessary after the affine transformation is taken into consideration in the optimization.

In terms of using GMM to compensate for noise, the method is similar to the compensation method proposed by Segura et. al. [5]. While the original Segura's method was only for additive noise, an extended method has been proposed by Fujimoto et. al. that compensates both additive and convolutional noises [6]. The differences are that while these methods apply compensation operation in the log spectral domain using both speech and noise models with some assumptions, our method compensates noise in the spectral domain using only speech GMM by directly maximizing its likelihood. Another extension of the Segura's method proposed by Miyake et. al. uses GMM likelihood as an object function to estimate SNR [7] but the difference is that their algorithm estimates only SNR rather than a general transformation.

The proposed method is also similar to feature space MLLR (constrained MLLR) [8, 9] in that it transforms input vectors based on model likelihood. A difference is where to apply the transformation. Feature space MLLR applies the transformation in the feature domain while the proposed method applies it in the spectral domain. Another difference is frequency resolution. For MFCC features, for example, the feature for speech recognition is derived from a filter bank output that has typically around 23 channels. The coarse frequency resolution is based on the observation that important information for speech recognition is encoded in spectral envelope rather than its finer structure and parsimony is useful in statistical modeling. However, from noise compensation point of view, the filter bank scatters the influence of noise across frequency as mentioned and spoils

the opportunity to remove noise that is originally localized. Our method applies the transformation to spectrum before the filter bank analysis. Therefore, it has a chance to compensate the noise effects without being affected by the feature extraction process. The time resolution is also different. While MLLR is performed using multiple utterances, our method estimates and applies a transformation for a short segment around 500 ms without requiring initial recognition hypothesis for the transformation estimation.

This paper is organized as follows. In Section 2, the algorithm of the proposed target speech GMM-based spectral compensation is described. Experimental conditions are shown in Section 3 and the results are presented in Section 4. Finally, a summary and future works are given in Section 5.

2. Target Speech GMM-based Spectral Compensation

In this section, we first describe how to formulate a spectral compensation transformation and then explain how the parameters of the transformation are optimized. We refer to our target speech GMM-based spectral compensation method as TGSC.

2.1. Spectral compensation transformation

Let ω be an index of a frequency bin of a spectral vector, x_ω be short time clean speech spectrum, a_ω be convolutional noise, and b_ω be additive noise. Then, noisy speech spectrum n_ω is denoted as Equation (1).

$$n_\omega = a_\omega \cdot x_\omega + b_\omega. \tag{1}$$

If a_ω and b_ω are known, clean speech spectrum is estimated using Equation (2).

$$\hat{x}_\omega = \frac{n_\omega}{a_\omega} - \frac{b_\omega}{a_\omega}. \tag{2}$$

In the following, we express the transformation more generally as $f(n_\omega, a_\omega, b_\omega)$ that transforms noisy speech n_ω depending on parameters a_ω and b_ω. The transformation is applied at each frequency bin independently. We use notations A and B to denote vectors consisting of a_ω and b_ω, respectively. The dimension of A and B is equal to the frequency resolution. For example, if speech waveform is sampled at 8 kHz and window size of an FFT analysis is 25 ms, then the dimension of A and B is $\frac{1}{2}8000 \cdot 0.025 = 100$. The problem is how to estimate A and B and we describe the proposed algorithm in the following.

2.2. Transformation parameter estimation

The proposed method uses a speech GMM that is estimated from the same training data as an acoustic model used for a decoding engine. Based on the GMM, the parameters A and B of the transformation are estimated for every block of input noisy speech spectrum vectors so as to maximize the likelihood of the transformed spectrum vectors $\{Y_1, Y_2, \cdots, Y_T\}$ as shown in Equations (3) and (4).

$$L(A,B) = \sum_{t=1}^{T} L_{GMM}(Y_t(A,B)), \tag{3}$$

$$\{A_{opt}, B_{opt}\} = \underset{A,B}{\operatorname{argmax}} \{L(A,B)\}, \tag{4}$$

where t is a frame index and $L_{GMM}(Y_t)$ is log likelihood of Y_t by the speech GMM. Since the likelihood evaluation is based on GMM, no state alignment is required and thus the process can be embedded in a feature extraction front-end. Note in the transformation estimation, the variables to be optimized are $\{A, B\}$, and the GMM is treated as a constant. The typical block size T that we assume is around 50, which corresponds to 500 ms when the frame rate is 100 Hz enabling very quick adaptation to noisy environments. The local optimum of Equation (4) can be obtained by using the gradient ascent method. The gradient is obtained by Equation (5) using the chain rule.

$$\frac{\partial L}{\partial p_\omega} = \sum_t \sum_k \frac{\partial L^t_{GMM}}{\partial y^t_k} \frac{\partial y^t_k}{\partial f^t_\omega} \frac{\partial f^t_\omega}{\partial p_\omega}, \tag{5}$$

where p_ω is either a_ω or b_ω, k is an index of an element of the feature vector Y_t, y^t_k is the k-th element of Y_t, and f^t_ω is the transformed spectrum.

For MFCC, the feature is derived from the (transformed) spectral vector by applying the filter bank analysis, log transformation, and discrete cosine transformation. Therefore, the gradient $\frac{\partial y_k}{\partial f_\omega}$ is obtained as Equation (6).

$$\frac{\partial y_k}{\partial f_\omega} = \frac{\partial}{\partial f_\omega} \sum_j c_{k,j} \log\left(\sum_{\omega'} w_{j,\omega'} \cdot f_{\omega'}\right), \tag{6}$$

where $c_{k,j}$ is (k,j) element of a discrete cosine transformation matrix C and $w_{j,\omega}$ is (j,ω) element of a filter bank matrix W. Since the delta coefficients [10] are linear sums of adjacent frames, their gradients are obtained as linear sums of gradients of the corresponding frames.

2.3. Implementation of transformation

Following the formulation of the magnitude spectrum subtraction [2], we assume magnitude spectrum rather than complex spectrum as the noisy speech spectrum n_ω. A problem is that after subtracting an estimated additive noise, the compensated magnitude spectrum might take a negative value. Therefore, a flooring operation is necessary as in the spectral subtraction and we formulated the spectral compensation transformation $f(n_\omega, a_\omega, b_\omega)$ as shown in Equation (7), in which n_ω, a_ω, and b_ω are all real numbers. The reason that we used a^2_ω and b^2_ω instead of a_ω and b_ω, respectively, was to limit the ranges of the multiplicative and subtractive compensation terms to n_ω non-negative during the optimization [1].

Since Equation (7) is non-continuous, the gradient method can not be directly applied. To make the optimization possible, we approximated Equation (7) by a continuous function as shown in Equation (8).

$$f_\omega = \max\left\{a^2_\omega \cdot n_\omega - b^2_\omega,\ 0.1 n_\omega\right\} \tag{7}$$

$$\approx log\left(exp\left(a^2_\omega \cdot n_\omega - b^2_\omega\right) + exp\left(0.1 n_\omega\right)\right). \tag{8}$$

As an initial values for a^2_ω and b^2_ω, proper constants may be used. Alternatively, if an estimate of an additive noise is known, it can be used as an initial value for b^2_ω. In that case, the proposed method reduces to spectral subtraction if $a^2_\omega = 1$ and the number of iterations for the gradient ascent is zero.

[1] While there are other choices to make the ranges non-negative such as $|a|$, taking a square gave the best result in our preliminary experiments.

Table 1: *SNR and Word accuracy. Base is a baseline, TGSCc is the proposed spectral compensation method initialized with a constant, SS is spectral subtraction, TGSCss is the proposed method initialized with a noise vector used in spectral subtraction. Bold results in TGSCc are better than Base and their differences are statistically significant. Similarly, bold results in TGSCss are better than SS and their differences are statistically significant.*

	SNR						
Method	clean	20	15	10	5	0	-5
Base	99.4	90.1	70.6	41.9	23.0	14.3	9.7
TGSCc	97.9	87.8	**71.7**	**47.6**	**25.5**	13.9	9.2
SS	98.7	92.2	83.7	66.5	41.6	21.8	11.6
TGSCss	**99.0**	**95.8**	**89.8**	**75.1**	**50.0**	**25.5**	**12.8**

3. Experimental setups

Speech recognition experiments were performed using the AURORA-2J database [11]. Both an HMM acoustic model for decoding and a GMM for the proposed noise compensation are estimated from the clean training data. The training data consisted of gender balanced 8440 utterances from 110 speakers. Test set was "test set C" of the AURORA-2J database whose channel condition is open to the training data. As additive noises, the test set includes subway and street noises with seven different SNRs. At each SNR condition, the test set has 2002 utterances. The recognition vocabulary was 13, which consisted of 11 entries for digits including two pronunciations for zero, and two silences with different lengths. The training and decoding were conducted following the scripts provided by the corpus. Therefore, the experiments corresponded to "category 0" according to the guideline of the corpus. The sampling frequency of the speech data was 8 kHz, and the window width and shift for the FFT analysis was 25 ms and 10 ms, respectively. Feature vectors consisted of 12 MFCC, their delta, and delta energy. Both the HMM and the GMM used these features.

For the spectral compensation by TGSC, 50 frames were treated as a unit to estimate and apply the transformation. As the initial value for a_ω^2, 1.0 was used. For b_ω^2, two settings were evaluated; one was a constant 100.0 and the other was a noise vector estimated from the first 10 frames of each utterance. When the estimated noise vector was used and if the parameters were not updated, TGSC gave the same result as the spectral subtraction as mentioned in Section 2.3.

4. Experimental results

Table 1 shows word accuracies for each SNR condition. The Gaussian mixture used for TGSC had 422 Gaussian components and the number of iterations for the gradient ascent to optimize the transformation was five. Compared to the baseline that applied no noise compensation, the proposed TGSC initialized with a constant vector (denoted as TGSCc in the table) gave some improvements when SNR was 5 to 10 but a slight degradation was observed in the clean condition. When a noise vector was used to initialize TGSC (denoted as TGSCss in the table), it gave better results than spectral subtraction for all the conditions and the differences were all statistically significant by the MAPSSWE significance test [12].

Figure 1 shows the relationship between the number of mixtures of GMM and word accuracy when SNR=10 and the number of iterations was five. Improvements from spectral subtraction were obtained when the number of mixtures was two or larger. The improvements became larger for the increase of the mixtures but it mostly converged at around 400 mixtures.

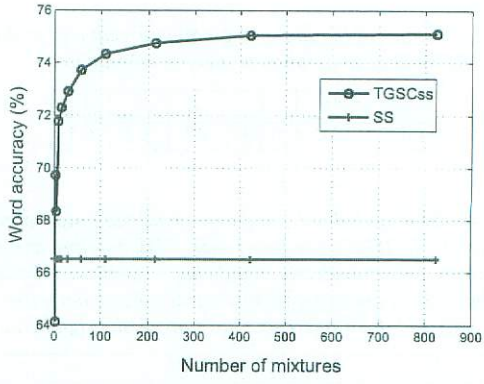

Figure 1: *Number of mixtures of GMM for TGSC and word accuracy when SNR=10. TGSCss is the result by the proposed method using an estimated noise vector as an initial value, and SS is a result by spectral subtraction that is independent of the GMM.*

Table 2: *Number of iterations of the gradient ascent to optimize TGSCss and word accuracy. Zero-th iteration is the result of spectral subtraction.*

	SNR						
# iter	clean	20	15	10	5	0	-5
0	98.7	92.2	83.7	66.5	41.6	21.8	11.6
1	98.8	94.1	86.6	71.0	46.5	23.8	12.5
2	99.0	94.9	88.2	72.9	48.9	24.9	13.1
5	99.0	95.8	89.8	75.1	50.0	25.5	12.8
10	98.3	95.8	89.6	74.0	48.0	23.5	12.1

Table 2 shows a relationship between the number of iterations and word accuracy when the noise vector was used for the initialization. In the table, the results of zero-th iteration are equivalent to spectral subtraction. The results of five iterations corresponds to the results of TGSCss in Table 1. As can be seen, improvements from the spectral subtraction method were obtained from the first iteration. Overall, five iterations gave the best results. The reason that 10 iterations gave slightly degraded results from five iterations was probably over-fitting.

Our software to apply TGSC was not optimized for efficient computation but Table 3 shows the real time factor (RTF) of TGSC using the program and 422 mixture Gaussian distribution on computers with an Intel Core 2 CPU. Since it involves the optimization of the transformation, TGSC is computationally more expensive than spectral subtraction. The cost was mostly linear to the number of iterations.

While TGSC can compensate convolutional noise, the normalization time scale is about 500 ms.To incorporate longer characteristics, cepstral mean subtraction (CMS) [13] would be useful. While there are many possibilities how to combine TGSC and CMS, here we simply applied them sequentially by first applying TGSC and then CMS. For this combination, the GMM used for TGSC was the same as that used in the experiments without CMS having 422 mixtures.Table 4 show the results in which the number of iterations for TGSC was five.

Table 3: *Number of iterations of gradient ascent and real time factor (RTF). Zero-th iteration is spectral subtraction.*

# iter	0	1	2	5	10
RTF	0.02	0.63	1.2	3.2	6.1

Table 4: *SNR and Word accuracy when CMS was applied. Bold results in TGSCc are better than CMS baseline and their differences are statistically significant. Similarly, bold results in TGSCss are better than spectral subtraction with CMS (SS+CMS) and their differences are statistically significant.*

	SNR						
Method	clean	20	15	10	5	0	-5
CMS	99.5	95.7	85.5	58.3	31.0	21.4	13.4
TGSCc	99.0	94.5	**86.5**	**67.4**	**44.4**	**26.0**	**15.5**
SS+CMS	99.1	94.6	89.6	77.4	56.0	31.1	15.5
TGSCss	**99.3**	**97.0**	**93.3**	**83.5**	**62.4**	**35.2**	**16.9**

Compared to Table 1, the baseline was replaced with the one with CMS that gave higher word accuracy. In this condition, TGSCc that used the constant for the initialization gave better results than the CMS baseline for SNR -5 to 15. A slight degradation was observed in the clean condition but it was relatively minor. When an estimated noise vector was used for the initialization, TGSC gave better results than spectral subtraction with CMS for all the conditions.

5. Conclusions

We have proposed a target speech GMM-based spectral compensation (TGSC) method for noise robust speech recognition. The proposed method uses the knowledge of speech sound as GMM and applies noise compensation transformation in the spectral domain. The transformation has a simple representation in the spectral domain addressing both additive and convolutional noises. The parameters of the transformation consisted of multiplicative and subtractive terms and they are optimized so that the transformed signal gives the highest GMM likelihood in the feature domain. The parameters are estimated for a block of frames whose length is around 50 frames or 500 ms. How to initialize the parameters of the transformation is important. In the experiment, when both the multiplicative and subtractive terms were initialized by constants, improvements from a baseline were observed depending on SNR. When the subtraction term was initialized by a noise vector, TGSC outperformed spectral subtraction at all the SNR conditions. A combination of TGSC and cepstrum mean subtraction (CMS) was also investigated by applying CMS to the output of TGSC, and it was shown that TGSC initialized with a noise vector gave the best results.

Future work includes the comparisons and combinations with other noise robustness or adaptation techniques such as MLLR. TGSC and MLLR have similarity in that the transforms are estimated based on the likelihood criterion. While an advantages of TGSC over MLLR is that it is more suitable for online decoding having shorter transformation estimation unit without requiring recognition hypothesis, it is interesting to combine TGSC and MLLR for offline processing since TGSC has higher frequency resolutions than MLLR. On the other hand, TGSC does not have ability to warp frequency axis that is useful to adapt to a speaker by normalizing the difference of vocal tract length which MLLR can do [14]. Therefore, an additive effect is expected by combining them. Since the performance of TGSC depends on how to initialize the parameters, more investigation on the initialization strategy will be useful. Reducing the computational cost for TGSC is necessary but we are also interested in investigating computationally more expensive variants for higher noise compensation performance by utilizing the emerging power of parallel processors such as GPGPU [15].

6. Acknowledgements

A sample program that implements the proposed algorithm will be available at `http://www.furui.cs.titech.ac.jp/~shinot/index.html`. This work was supported by KAKENHI (21700188).

7. References

[1] M. Gales and S. Young, "The application of hidden markov models in speech recognition," *Foundations and Trends in Signal Processing*, vol. 1, no. 3, pp. 195–304, 2008.

[2] S. F. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Transactions on Acoustics Speech and Signal Processing*, vol. 27, no. 2, pp. 113–120, 1979.

[3] A Sankar and C. H. Lee, "A maximum likelihood approach to stochastic matching for robust speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 4, no. 3, pp. 190–202, 1996.

[4] D. Kim and D. Yook, "Feature transform in linear spectral domain for fast channel adaptation," *Electronics Letters*, vol. 40, no. 20, pp. 1313–1314, 2004.

[5] J. C. Segura, M. C. Benitez A de la Torre, and A. M. Peinado, "Model-based compensation of the additive noise for continuous speech recognition. experiments using the Aurora II database and tasks," in *Proc. Eurospeech*, 2001, pp. 221–224.

[6] M. Fujimoto and Y. Ariki, "Robust speech recognition in additive and channel noise environments using GMM and EM algorithm," in *Proc. ICASSP*, 2004, vol. I, pp. 941–944.

[7] N. Miyake, T. Takiguchi, and Y. Ariki, "Sudden noise reduction based on gmm with noise power estimation," in *Proc. Interspeech*, 2008, pp. 403–406.

[8] V. Digalakis, D. Rtischev, L. Neumeyer, and Edics Sa, "Speaker adaptation using constrained estimation of Gaussian mixtures," *IEEE Transactions on Speech and Audio Processing*, vol. 3, pp. 357–366, 1995.

[9] M.J.F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, pp. 75–98, 1998.

[10] S. Furui, "Cepstral analysis technique for automatic speaker verification," *IEEE Transactions on Acoustics Speech and Signal Processing*, vol. 29, no. 2, pp. 254–272, 1981.

[11] S. Nakamura et.al., "AURORA-2J: An evaluation framework for Japanese noisy speech recognition," *IEICE Transactions on Information and Systems*, vol. E88-D, no. 3, pp. 535–544, 2005.

[12] L. Gillick and S. Cox, "Some statistical issues in the comparison of speech recognition algorithms," in *Proc. ICASSP*, 89, pp. 532–535.

[13] B. Atal, "Effectiveness of linear prediction characteristics of of the speech wave for automatic speaker identification and verification," *The Journal of the Acoustical Society of America*, vol. 55, no. 6, pp. 1304–1312, 1974.

[14] G. H. Ding, Y. F. Zhu, C. Li, and B. Xu, "Implementing vocal tract length normalization in the mllr framework," in *Proc. ICSLP*, 2002, pp. 1389–1392.

[15] E. Wu and Y. Liu, "Emerging technology about GPGPU," in *Proc. APCCAS*, 2008, pp. 618–622.