Eigen-Environment based Noise Compensation Method for Robust Speech Recognition

Hwa Jeon Song and Hyung Soon Kim

Department of Electronics Engineering,
Pusan National University, Korea
{hwajeon, kimhs}@pusan.ac.kr

Abstract

In this paper, we propose a new noise compensation method based on the eigenvoice framework in feature space to reduce the mismatch between training and testing environments. In this method, the difference between clean and noisy environments is represented by the linear combination of K eigenvectors that represent the variation among environments. Since how to construct the noisy models is crucial for the performance of the proposed method, we introduce two methods for constructing noisy models : one based on MAP adaptation method and the other using stereo DB. In experiments using Aurora 2 DB, we obtained 44.9% relative improvement with eigen-environment method in comparison with baseline system. Especially, in clean condition training mode, our proposed method yielded 67.4% relative improvement.

1. Introduction

When there is a mismatch between training and testing environments, ASR systems suffer degradation in performance. Though several kinds of methods to alleviate this mismatch have been presented, the problem in ASR system still remains unsolved. Major sources of this mismatch include the variations of speaking environments and of speakers. Among them, the variations of speaking environment include ambient noises, communication channel distortions, transducer distortions, and so on. Various additive noises and convolutional noises are primary factors of changes in the speaking environment. Several kinds of methods to remove these noisy components have been proposed in feature space.

Recently, Aurora project [1] to evaluate the performance of the distributed speech recognizer (DSR) in noisy environment using the common DB has been progressed. The compensation method proposed by [2] has been accepted as Advanced front-end[3] among many proposed methods. And Stereo-based Piecewise LInear Compensation for Environments (SPLICE) method[4] based on stereo DB also showed a good performance and related methods with SPLICE has been developed[5][6].

In this paper, a new noise compensation method based on the eigenvoice framework in feature space is proposed to reduce the mismatch between training and testing environments. The difference between clean and noisy environments in the proposed method can be represented by the linear combination of K eigenvectors that contain the most important components of the variations of various noisy environments from the clean environment. The main factor for performance improvement of ASR systems in proposed method is how to construct the noisy models and the set of bias vectors of each model. In this paper, two methods, one using non-stereo data the other using stereo data, are proposed to construct the noisy models and the set of bias vectors.

In section 2, we propose a new noise compensation method in feature space based on eigenvoice adaptation framework. We call it *eigen-environment* method. In section 3, our proposed method is compared with baseline system on Aurora 2 DB. Finally, section 4 provides conclusions and future works.

2. Noise Compensation based on Eigen-Environment

2.1. Overview of Eigen-Environment

The eigen-environment based noise compensation method proposed in this paper is briefly shown in Fig. 1. The noise compensation procedure in eigen-environment approach is very similar to that of eigenvoice adaptation method[7]. First, in off-line step, R GMMs for R noisy environments and a set of bias compensation vectors for indicating the gap between clean and noisy environment to obtain the prior information of variability among several environments are constructed. And the bias supervectors are obtained by concatenating M mean vectors included in each GMM.

We call the space spanned by these supervectors as the environment space. It should be noted that the components in the same dimension among supervectors must have high correlation. The total number of dimensions of supervector L is the number of Gaussian mixtures (M) multiplied by the number of dimension of observation vector (D).

Then, R eigenvectors with dimension L are made by adopting PCA to reduce the dimensionality. We call these eigenvectors eigen-environments. If the K principal components (or eigenvectors) can cover most of the environment space, L dimensional vector $\mathbf{x}$ in environment space can be represented by the weighted sum of K eigenvectors or eigen-environments as follows:

$$\mathbf{x} \cong \mathbf{e}(0) + \sum_{k=1}^{K} w(k)\mathbf{e}(k) \tag{1}$$

where $\mathbf{e}(0)$ and $\mathbf{e}(k)$ denote the mean supervector of R bias supervectors and k-th eigen-environment, respectively. And $w(k)$ is the weight on k-th eigen-environment.

In the on-line step as similar to the eigenvoice method, the input noisy test speech can be compensated with (1). The noisy models and bias vector sets, constructed previously in the off-line step, can be used to estimate the weights. That is, they can be used to find the posteriori probability and to select the environment.

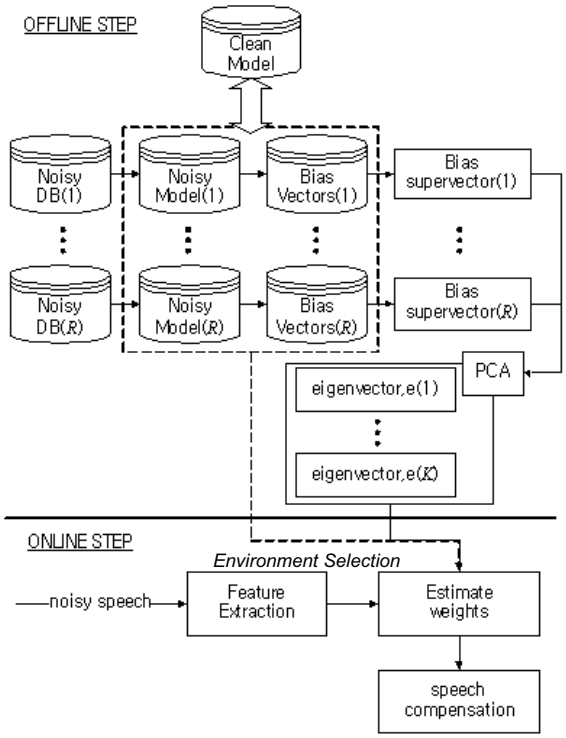

Figure 1: diagram of eigen-environment

2.2. Weight Estimation of Eigen-Environment

First, define some notations used in eigen-environment approach.

$\mathbf{X} = \{\mathbf{x}_1, \cdots, \mathbf{x}_T\}$: clean speech data

$\mathbf{Y} = \{\mathbf{y}_1, \cdots, \mathbf{y}_T\}$: noisy speech data

Assume that $\mathbf{X}$ and $\mathbf{Y}$ are simultaneously recorded to train GMMs and the relationship between them is represented as follows:

$$\mathbf{y} = \mathbf{x} + \mathbf{g}(\mathbf{x}, \mathbf{h}, \mathbf{n}) \tag{2}$$

and

$$\mathbf{g}(\mathbf{x}, \mathbf{h}, \mathbf{n}) = \mathbf{h} + \mathbf{D}\ln(\mathbf{I} + \exp[\mathbf{D}^T(\mathbf{n} - \mathbf{h} - \mathbf{x})]) \tag{3}$$

where $\mathbf{y}$, $\mathbf{x}$, $\mathbf{h}$ and $\mathbf{n}$ are the cepstrum vector of noisy speech, clean speech, channel distortion and the additive noise, respectively. $\mathbf{g}(\cdot)$ is the non-linear function of noisy components and clean speech in cepstrum domain. And matrix $\mathbf{D}$ denotes Discrete Cosine Transformation(DCT) matrix.

Let the basic formulation set up in order to derive the estimation of the weight of eigen-environment. First, assume that noisy speech is modeled with M Gaussian mixtures as follows:

$$p(\mathbf{y}) = \sum_{m=1}^{M} p(\mathbf{y} \mid m)p(m) = \sum_{m=1}^{M} N(\mathbf{y} \mid \boldsymbol{\mu}_{\mathbf{y}}^m, \boldsymbol{\Sigma}_{\mathbf{y}}^m)p(m) \tag{4}$$

where $p(m)$, $\boldsymbol{\mu}_{\mathbf{y}}^m$ and $\boldsymbol{\Sigma}_{\mathbf{y}}^m$ are prior probability, mean vector and covariance matrix of m-th Gaussian mixture, respectively. If clean speech $\mathbf{x}$ and noisy speech $\mathbf{y}$ have the relationship of joint Gaussian in mixture m, $p(\mathbf{x} \mid \mathbf{y}, m)$ is the Gaussian distribution with the mean vector given by

$$\begin{aligned} E[\mathbf{x} \mid \mathbf{y}, m] &= \boldsymbol{\mu}_{\mathbf{x}}^m + \boldsymbol{\Sigma}_{\mathbf{xy}}^m (\boldsymbol{\Sigma}_{\mathbf{y}}^m)^{-1}(\mathbf{y} - \boldsymbol{\mu}_{\mathbf{y}}^m) \\ &= \boldsymbol{\Sigma}_{\mathbf{xy}}^m (\boldsymbol{\Sigma}_{\mathbf{y}}^m)^{-1}\mathbf{y} + (\boldsymbol{\mu}_{\mathbf{x}}^m - \boldsymbol{\Sigma}_{\mathbf{xy}}^m (\boldsymbol{\Sigma}_{\mathbf{y}}^m)^{-1}\boldsymbol{\mu}_{\mathbf{y}}^m) \\ &= \mathbf{C}_m \mathbf{y} + \mathbf{r}_m \end{aligned} \tag{5}$$

where $\mathbf{C}_m = \boldsymbol{\Sigma}_{\mathbf{xy}}^m (\boldsymbol{\Sigma}_{\mathbf{y}}^m)^{-1}$, $\mathbf{r}_m = \boldsymbol{\mu}_{\mathbf{x}}^m - \boldsymbol{\Sigma}_{\mathbf{xy}}^m (\boldsymbol{\Sigma}_{\mathbf{y}}^m)^{-1}\boldsymbol{\mu}_{\mathbf{y}}^m$, and $\boldsymbol{\mu}_{\mathbf{x}}^m$ are the rotation matrix, the compensation vector and mean vector of a clean speech $\mathbf{x}$ in mixture m, respectively. And $\boldsymbol{\Sigma}_{\mathbf{xy}}^m$ denotes the cross-covariance matrix between $\mathbf{x}$ and $\mathbf{y}$. The compensation vector $\mathbf{r}_m$ is also called as the correction vector or bias vector. Hence, the estimated clean speech $\mathbf{x}$ based on minimum mean square error (MMSE) criterion given noisy speech $\mathbf{y}$ is as follows:

$$\hat{\mathbf{x}}_{MMSE} = E[\mathbf{x} \mid \mathbf{y}] = \sum_m p(m \mid \mathbf{y})E[\mathbf{x} \mid \mathbf{y}, m] \tag{6}$$

where

$$p(m \mid \mathbf{y}) = \frac{p(\mathbf{y} \mid m)p(m)}{\sum_m p(\mathbf{y} \mid k)p(m)} \tag{7}.$$

If (5) is inserted in (6) and the rotation matrix $\mathbf{C}_m$ in (5) is assumed as the identity matrix, then (6) is simplified as follows:

$$\hat{\mathbf{x}}_{MMSE} = \mathbf{y} + \sum_m p(m \mid \mathbf{y})\mathbf{r}_m \tag{8}$$

And the compensation vector in (5) is also simplified as follows:

$$\mathbf{r}_m = \boldsymbol{\mu}_{\mathbf{x}}^m - \boldsymbol{\mu}_{\mathbf{y}}^m \tag{9}$$

Therefore, the estimation on the clean speech in (8) can be explained as an estimation of the posteriori probability and the correction vector on each mixture given noisy speech.

Most of compensation methods based on GMM[8]-[10] have different ways of estimating $\mathbf{r}_m$ in (9). These methods can be categorized into on-line and off-line estimation methods. Especially, the off-line methods can be divided into two categories: one using the stereo DB, the other non-stereo DB. In SPLICE or stereo-based RATZ [9], the compensation vector $\mathbf{r}_m$ of each mixture is previously constructed by using stereo DB in off-line step while in blind RATZ[9], the compensation vector is constructed by using non-stereo DB. In vector Taylor series(VTS) approach[9], noisy speech is

compensated as to estimate terms related with $\boldsymbol{\mu}_{\mathbf{y}}^{m}$ in on-line step.

In this paper, based on the eigen-environment concept in section 2.1, we propose the on-line estimation method for finding the compensation vector $\mathbf{r}_m$ by the weighted sum of eigen-environments as follows:

$$\mathbf{r}_m = \mathbf{e}_m(0) + \sum_{k=1}^{K} w(k)\mathbf{e}_m(k) \tag{10}$$

where $\mathbf{e}_m(0)$ and $\mathbf{e}_m(k)$ are the mean subvector of bias supervectors and k-th eigen-environment corresponding to m-th Gaussian mixture, respectively. To estimate the weights in (10), we also use EM algorithm based on Q-function as

$$Q(\lambda, \hat{\lambda}) = -\frac{1}{2} P(\mathbf{Y} \mid \lambda) \sum_{m} \sum_{t} p(m \mid \mathbf{y}_t) f(\mathbf{y}_t, m) \tag{11}$$

where $\mathbf{Y}$ and $\mathbf{y}_t$ are a test noisy speech sequence and a frame in $\mathbf{Y}$ at time t, respectively. And

$$f(\mathbf{y}_t, m) = D\log(2\pi) + \log|\boldsymbol{\Sigma}_{\mathbf{y}}^{m}| + h(\mathbf{y}_t, m) \tag{12}$$

where D is the dimensionality of noisy feature vector $\mathbf{y}_t$ and

$$h(\mathbf{y}_t, m) = (\mathbf{y}_t - \boldsymbol{\mu}_{\mathbf{y}}^{m})^T (\boldsymbol{\Sigma}_{\mathbf{y}}^{m})^{-1} (\mathbf{y}_t - \boldsymbol{\mu}_{\mathbf{y}}^{m}) \tag{13}$$

The mean vector of noisy speech $\boldsymbol{\mu}_{\mathbf{y}}^{m}$ in (13) is obtained by using the relationship between (9) and (10).

$$\boldsymbol{\mu}_{\mathbf{y}}^{m} = \boldsymbol{\mu}_{\mathbf{x}}^{m} + \mathbf{e}_m(0) + \sum_{k=1}^{K} w(k)\mathbf{e}_m(k) \tag{14}$$

(14) can be substituted for $\boldsymbol{\mu}_{\mathbf{y}}^{m}$ in (13) and $Q(\cdot)$ function is differentiated to find the maximum value by each weight. As a result, we can obtain the same equation with the solution of Maximum Likelihood Eigen-Decomposition(MLED) method[10] in eigenvoice adaptation framework. If the weights are solved from the equation, the compensated clean speech is represented as

$$\begin{aligned} \hat{\mathbf{x}}_{MMSE} &= \mathbf{y} + \sum_{m} p(m \mid \mathbf{y}) \hat{\mathbf{r}}_m \\ &= \mathbf{y} + \sum_{m} p(m \mid \mathbf{y}) \left(\mathbf{e}_m(0) + \sum_{k=1}^{K} \hat{w}(k)\mathbf{e}_m(k) \right) \end{aligned} \tag{15}$$

where $\hat{\mathbf{r}}_m$ and $\hat{w}(k)$ are the estimated compensation vector and the weight of k-th eigen-environment, respectively.

Although the proposed method has the off-line step to construct the eigen-environments, it is similar to VTS method in that the bias compensation vector is not fixed but variable with time while SPLICE or RATZ method is operated with the fixed bias compensation vectors. Therefore, we can expect that our proposed method compensates for a noisy speech more effectively than SPLICE.

2.3. Construction of Noisy Models

The important part in proposed method is how to construct the noisy models and bias vector sets. In this paper, we introduce two approaches to construct the model and bias compensation vector set. One is based on MAP adaptation using non-stereo DB and the other is based on the approach by using stereo DB. First, in MAP adaptation based method, GMM on noisy data corresponding to each noise type is constructed by MAP adaptation from GMM for clean DB. This approach does not require stereo DB. And the bias vector set is built by using the difference between clean and noisy means using (9).

The other method using stereo DB is that the clean GMM is built by the clean DB and the forced alignment is applied to obtain the mixture index of GMM for clean DB as follows:

$$\hat{m}_t = \arg\max_{m} P(\mathbf{x}_t \mid M, \boldsymbol{\Lambda}_{\mathbf{x}}) \tag{16}$$

The mixture with the highest probability among M mixtures is selected. As using stereo DB, noisy frames can also use the same mixture index information obtained from clean DB. The clustering module operates to construct the noisy GMM and bias vector set. We can build the each noisy GMM as to take the expectation for noisy speech frames included in the same mixture with the corresponding clean speech frame as follows:

$$\boldsymbol{\mu}_{m}^{\mathbf{y}} = E[\mathbf{y}_t \mid \hat{m}_t = m] \tag{17}$$

And the bias vector set is constructed by taking the mean of differences between noisy and clean speech frames as follows:

$$\mathbf{r}_m = E[\mathbf{x}_t - \mathbf{y}_t \mid \hat{m}_t = m] \tag{18}$$

3. Experiments and Results

For the evaluation of the proposed method in this paper, we conducted the experiments using Aurora 2 DB. Training and testing procedure follows the recommendation of [1].

The results of the proposed method are shown in Table 1. In our proposed eigen-environment method, noisy models are used to select an environment, just as in SPLICE. Although the improvement rate of method based on non-stereo data to build the noisy models and bias vector set does not reach the level of the method based on stereo data, that method also shows a performance improvement in comparison with that of baseline system. However, if the index sequence of environment extracted from clean test data can be used as a good estimation of the posteriori probability of noisy frames as an upper limit experiment, non-stereo data based eigen-environment shows the extremely improved performance in Table 1-(c). Therefore, we can see that how to build the noisy model based on eigen-environment method has a major influence on the performance of system.

The experiments of eigen-environment according to the analysis window size for real-time processing are conducted and those results are shown in Table 2. Although the experiment using the global window shows the best performance, the difference of performance among them is negligible. Therefore, the proposed method can be also applied to real-time processing. And in case of the same training and testing conditions such as [1], the proposed method shows better performance than other previously proposed methods such as [2] and [5] when SNR is in low level. That is, we can see that our proposed method is more robust in very noisy environment than other methods.

Table 1: Performance of proposed method. (a) using non-stereo data (b) using stereo data (c) upper limit (non-stereo data)

(a)

Absolute performance				
Training Mode	Set A	Set B	Set C	Overall
Multicondition	88.37	87.32	89.54	88.18
Clean Only	74.79	78.60	75.08	76.37
Average	81.58	82.96	82.31	**82.28**

Performance relative to Mel-cepstrum				
Training Mode	Set A	Set B	Set C	Overall
Multicondition	4.52%	7.66%	35.52%	13.18%
Clean Only	34.78%	51.64%	26.40%	40.83%
Average	19.65%	29.65%	30.96%	**27.00%**

(b)

Absolute performance				
Training Mode	Set A	Set B	Set C	Overall
Multicondition	90.01	88.52	90.53	89.52
Clean Only	86.84	86.46	87.00	86.72
Average	88.43	87.49	88.77	**88.12**

Performance relative to Mel-cepstrum				
Training Mode	Set A	Set B	Set C	Overall
Multicondition	18.05%	16.35%	41.64%	22.99%
Clean Only	65.95%	69.39%	61.60%	66.74%
Average	42.00%	42.87%	51.62%	**44.86%**

(c)

Absolute performance				
Training Mode	Set A	Set B	Set C	Overall
Multicondition	97.86	97.89	97.81	97.86
Clean Only	98.42	98.51	98.43	98.46
Average	98.14	98.20	98.12	**98.16**

Performance relative to Mel-cepstrum				
Training Mode	Set A	Set B	Set C	Overall
Multicondition	82.45%	84.65%	86.50%	84.30%
Clean Only	95.91%	96.63%	95.35%	96.13%
Average	89.18%	90.64%	90.93%	**90.22%**

Table 2: Performance comparison according to the size of analysis window when the number of eigen-environments is 15. (C) in windows size column means that the value of clean bias is constrained to be zero.

Window Size	Multi-condition		Clean-condition		Average Improve-ment (%)
	Word Accuracy (%)	Relative Improve-ment (%)	Word Accuracy (%)	Relative Improve-ment (%)	
50ms(C)	89.20	20.63	86.59	66.42	43.52
100ms(C)	89.33	21.59	86.84	67.06	44.32
150ms(C)	89.36	21.82	86.85	67.07	44.45
Global(C)	89.39	22.05	86.97	67.38	44.71
Global	89.52	22.99	86.72	66.74	44.86

4. Conclusions

In this paper, we proposed a new noise compensation method in the environment space based on eigenvoice adaptation framework. We call it *eigen-environment* method. In experiments using Aurora 2 DB, we obtained 45.03% relative improvement of eigen-environment method in comparison with baseline system when the number of eigen-environments is six. Especially, our proposed method shows 67.38% relative performance improvement rate in clean-condition mode. It is better performance than several algorithms previously proposed in Aurora project such as SPLICE and advanced front-end in clean-condition training mode. As future works, we will concentrate on the improvement of the performance in multi-condition training mode and the performance of the case using non-stereo data.

This paper was performed for the Intelligent Robotics Development Program, one of the 21st Century Frontier R&D Programs funded by the Ministry of Commerce, Industry and Energy of Korea.

5. References

[1] H. G. Hirsch and D. Pearce, "The AURORA experimental framework for the performance evaluations of speech recognition systems under noisy conditions," ISCA ITRW AST2000 "Automatic Speech Recognition: Challenges for the Next Millennium," Paris, France, Sep. 2000.

[2] D. Macho, L. Mauuary, B. Noe, Y. M. Cheng, D. Ealey, D. Jouvet, H. Kelleher, D. Pearce, F. Saadoun, "Evaluation of a noise-robust DSR front-end on Aurora databases," Proc. ICSLP, Denver, pp.17-20. Sep. 2002.

[3] ETSI standard document, "Speech processing, Transmission and Quality aspects (STQ) ; Distributed speech recognition; Advanced front-end feature extraction algorithm; Compression algorithms," ESTI ES 202 050 v1.1.1 (2002-10), Oct. 2002.

[4] L. Deng, A. Acero, M. Plumpe and X. Haung, "Large vocabulary continuous speech recognition under adverse conditions," Proc. ICSLP, Beijing, vol.3, pp.806-809, Oct. 2000.

[5] J. Droppo, L. Deng and A. Acero, "Evaluation of the SPLICE algorithm on the Aurora 2 database," Proc. Eurospeech, pp.217-220, Sep. 2001.

[6] J. Droppo, A. Acero and L. Deng, "Efficient on-line acoustic environment estimation for FCDCN in a continuous speech recognition system," Proc. ICASSP, vol.1, pp.209-212, May 2001.

[7] R. Kuhn, P. Nguyen, J. C. Jungua, L. Goldwasser, N. Niedzielski, S. Finche, K. Field and M. Contolini, "Eigenvoices for speaker adaptation," Proc. ICSLP, vol.5, pp.1771-1774, Nov. 1998.

[8] A. Acero, *Acoustical and Environmental Robustness in automatic speech recognition*, Ph.D. Thesis, Department of Electrical and Computer Engineering, Carnegie Mellon University, Sep. 1990.

[9] P. Moreno, B. Raj and R.M. Stern, "A unified approach to robust speech recognition," Proc. Eurospeech, Madrid, pp.480-484, Sep. 1995.

[10] P. J. Moreno, B. Raj and R. M. Stern, "A vector Taylor series approach for environment-independent speech recognition," Proc. ICASSP, vol.1, pp.733-736, May 1996.