2012 International Workshop on Nonlinear Circuits,
Communications and Signal Processing
NCSP'12, Honolulu, Hawaii, USA. March 4-6, 2012

Combination of SPLICE and Feature Normalization for Noise Robust Speech Recognition

Tsunenobu Kai, Masayuki Suzuki, Keigo Chijiiwa, Nobuaki Minematsu, Keikichi Hirose

The University of Tokyo, 7-3-1, Hongo, Bunkyo-ku, Tokyo, 113-0033, Japan
{kai,suzuki,keigo,mine,hirose}@gavo.t.u-tokyo.ac.jp

Abstract

It is well-known that the performance of automatic speech recognition (ASR) systems are easily affected by acoustic mismatch between training and testing conditions. This mismatch is often caused by various kinds of environmental noise or distortion. To reduce the effect of mismatch, feature normalization, feature enhancement, model adaptation, etc. have been studied intensively. Cepstral mean normalization (CMN), mean and variance normalization (MVN) and histogram equalization (HEQ) are well-known methods of feature normalization. Stereo-based piecewise linear compensation for environments (SPLICE) is one of the feature enhancement methods. In this paper, we describe how to combine these methods to effectively improve the robustness of ASR systems. In the experiments performed on the Aurora-2 database, a good combination showed a 41% improvement in word error rate over SPLICE only, and a 25% improvement over the conventional combination of SPLICE and CMN.

1. Introduction

When there is a mismatch between the acoustic conditions of training acoustic models and using them in applications, the performance of a speech recognition system is often seriously degraded. Various sources give rise to this mismatch, such as background noise, acoustic characteristics of recording devices, channel distortion, etc. Compensation methods for robust ASR mainly focus on minimizing this mismatch.

Feature normalization methods are applied as a part of the feature extraction in order to minimize the mismatch. CMN [1] and MVN [2] are well-known methods of feature normalization. CMN removes the average value of the feature vector from each observation. This normalization compensates for the main effect of channel distortion. MVN normalizes not only the average but also the variance of the feature vectors. HEQ, which is a feature normalization method frequently used in digital image processing, is also efficient in speech recognition [3, 4]. HEQ transforms acoustic features so that the histogram of transformed features resembles the normal distribution. Because the transformation can be non-linear, HEQ can compensate for non-linear distortion by noise.

SPLICE [5] is well known as a frame-based noise removal algorithm for feature enhancement. SPLICE approximates the non-linear transformation from noisy features to its clean version by probabilistic summation of piecewise linear transformations. The weights of transformations are calculated by using a Gaussian mixture model (GMM) of noisy features. Because transformations and a GMM of noisy features are trained in advance, the enhancement procedure of SPLICE requires a low computational cost but the performance of SPLICE is high. However, the performance of SPLICE is poor in a different environment from training one.

Although each method can reduce the mismatch to some degrees, as far as we know, it seems that not so much attention was paid to how to combine these methods. In this paper, we investigate different combinations for more robust ASR. Application of feature normalization after SPLICE is expected to reduce the distortion that still remains after SPLICE. Considering that SPLICE accepts any type of features as its input, normalized features can be used as input to SPLICE to enhance them. Because normalized features contain less mismatches than original features, the enhancement performance of SPLICE is expected to improve.

2. Methods

In this section, we briefly explain some methods to improve the robustness against noise. In particular, we introduce two normalization methods, CMN and HEQ, and a feature enhancement method, SPLICE.

2.1. Feature normalization

Cepstral Mean Normalization (CMN) subtracts the average of cepstra from themselves. Since cepstra are derived from log spectra, CMN has the effect of reducing sensitivity to channel distortion. The normalized feature $\hat{\boldsymbol{x}}$ is

$$\hat{\boldsymbol{x}} = F(\boldsymbol{x}) = \boldsymbol{x} - \boldsymbol{\mu}\ , \tag{1}$$

where $\boldsymbol{\mu}$ is the mean of the original feature $\boldsymbol{x}$. CMN makes the mean of the normalized feature $\hat{\boldsymbol{x}}$ zero and so equalizes

the first moment of its probability distribution. CMN is particularly simple but can realize robust speech recognition.

Histogram Equalization (HEQ) is a feature normalization method to provide a transformation F from $\boldsymbol{x}$ to $\hat{\boldsymbol{x}}$. F is calculated as

$$\hat{\boldsymbol{x}} = F(\boldsymbol{x}) = C_{\mathrm{normal}}^{-1}(C(\boldsymbol{x})) \ , \tag{2}$$

where C is the cumulative distribution of the original feature $\boldsymbol{x}$ and C_{normal}^{-1} is the inverse function of the cumulative distribution of the standard normal distribution. HEQ makes the histogram of the normalized feature $\hat{\boldsymbol{x}}$ the standard normal distribution. In other words, HEQ equalizes all the moments of the probability distribution to those of the standard normal distribution. For this reason, HEQ can be considered as an extension of CMN or MVN. Because the transformation F is non-linear, HEQ can compensate for non-linear distortion.

2.2. SPLICE

SPLICE approximates the non-linear transformation from noisy feature $\boldsymbol{y}$ to its clean version $\boldsymbol{x}$ by probabilistic summation of piecewise linear transformations. We obtain an estimate $\hat{\boldsymbol{x}}$ of the clean feature $\boldsymbol{x}$ as

$$\hat{\boldsymbol{x}} = \sum_k p(k|\boldsymbol{y}) \boldsymbol{A}_k \boldsymbol{y'} \ , \tag{3}$$

where $\boldsymbol{A}_k$ is a linear transformation and $\boldsymbol{y}'$ is an augmented feature vector given by $[1 \ \ \boldsymbol{y}^\top]^\top$. $\boldsymbol{A}_k$ is trained in advance by using stereo data and $p(k|\boldsymbol{y})$ is calculated by using GMM of noisy features. k is an index of the GMM component.

In a training step of SPLICE, we firstly characterize a probability density function of noisy features $\boldsymbol{y}$ as GMM

$$p(\boldsymbol{y}) = \sum_k \pi_k \mathcal{N}(\boldsymbol{y}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \ , \tag{4}$$

where π_k, $\boldsymbol{\mu}_k$, $\boldsymbol{\Sigma}_k$ are the weight, the average, the variance of the k-th component. Next, we estimate the linear transformation $\boldsymbol{A}_k$ based on the weighted minimum mean square error criterion.

$$\boldsymbol{A}_k = \underset{\boldsymbol{A}_k}{\mathrm{argmin}} \sum_i p(k|\boldsymbol{y}_i) ||\boldsymbol{x}_i - \boldsymbol{A}_k \boldsymbol{y}_i'||^2 \ . \tag{5}$$

This estimation needs stereo data, namely, noisy features $\boldsymbol{y}_i$ and their clean version $\boldsymbol{x}_i$. Because transformations $\boldsymbol{A}_k$ and a GMM of noisy features $\boldsymbol{y}$ are trained in advance, the enhancement procedure of SPLICE requires a low computational cost but the performance of SPLICE is high. However, since $\{\boldsymbol{A}_k\}$ are trained using stereo data in the training dataset only, the performance of SPLICE has to be poor in a different environment from training one.

2.3. Combination of HEQ and SPLICE

In the literature [5], a method to apply CMN after SPLICE was proposed, which will be called SPLICE-CMN. In this paper, we propose more robust features using HEQ after SPLICE, SPLICE-HEQ, obtained by the following formula:

$$\hat{\boldsymbol{x}} = \sum_k p(k|\boldsymbol{y}) \boldsymbol{A}_k \boldsymbol{y'} \tag{6}$$

$$\hat{\hat{\boldsymbol{x}}} = C_{\mathrm{normal}}^{-1}(C(\hat{\boldsymbol{x}})) \ , \tag{7}$$

where $\hat{\boldsymbol{x}}$ is estimated feature by SPLICE. HEQ is expected to compensate for non-linear distortion which CMN cannot deal with well.

Moreover, SPLICE can take any type of features as input and transform them adequately [6]. So, we propose to apply HEQ to noisy features beforehand and input the resulting features to SPLICE, HEQ-SPLICE. The final feature $\hat{\hat{\boldsymbol{x}}}$ is

$$\hat{\boldsymbol{y}} = C_{\mathrm{normal}}^{-1}(C(\boldsymbol{y})) \tag{8}$$

$$\hat{\hat{\boldsymbol{x}}} = \sum_k p(k|\hat{\boldsymbol{y}}) \boldsymbol{A}_k \hat{\boldsymbol{y}}' \ , \tag{9}$$

where $\hat{\boldsymbol{y}}$ is a normalized noisy feature by HEQ. A GMM of $\hat{\boldsymbol{y}}$ and $\boldsymbol{A}_k$ are trained using normalized clean feature $\hat{\boldsymbol{x}}_i$ and normalized noisy feature $\hat{\boldsymbol{y}}_i$ in advance according to the following formulae:

$$p(\hat{\boldsymbol{y}}) = \sum_k \pi_k \mathcal{N}(\hat{\boldsymbol{y}}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{10}$$

$$\boldsymbol{A}_k = \underset{\boldsymbol{A}_k}{\mathrm{argmin}} \sum_i p(k|\hat{\boldsymbol{y}}_i) ||\hat{\boldsymbol{x}}_i - \boldsymbol{A}_k \hat{\boldsymbol{y}}_i'||^2 \ . \tag{11}$$

Because normalized features are supposed to contain less mismatches than original features, SPLICE is expected to enhance input features more adequately.

3. Experimental results

Experiments of various combinations were conducted on the Aurora-2 database [7]. This database consists of English connected digit utterances in the presence of additive noise and linear convolutional distortion. The database prepares three test sets to measure the ASR performance. Set A contains noises similar to those found in the training data, set B contains new additive noises, and set C contains new convolutional distortion. We used HMMs which are 16-states whole-word models for each digit and have 20 diagonal Gaussian mixture components in each state. We prepared two types of HMMs. The first HMMs were trained by using clean speech data only (clean acoustic models), and the second HMMs were trained by using both clean speech data and noisy speech data (multi-conditions models). We used MFCC+Δ+$\Delta\Delta$ totaling 39 dimensions as feature vector. We investigated

Table 1: Summary of word accuracies for HEQ using clean acoustic models

HEQ	Set A				Set B				Set C		
	N1	N2	N3	N4	N1	N2	N3	N4	N1	N2	Average
CLEAN	99.66	99.70	99.46	99.63	99.66	99.70	99.46	99.63	99.69	99.64	99.62
SNR20	98.04	98.61	98.81	97.59	98.96	97.97	98.81	98.33	98.19	98.22	98.35
SNR15	95.64	96.49	97.05	94.91	96.90	96.10	97.29	96.02	95.46	95.98	96.18
SNR10	88.70	91.05	91.83	87.84	92.05	90.39	93.20	91.08	89.35	90.02	90.55
SNR5	74.95	73.88	75.22	73.28	77.22	75.24	78.17	76.18	73.87	76.39	75.44
SNR0	46.48	43.05	46.47	47.49	50.02	45.95	51.21	47.27	45.96	46.52	47.04
SNR-5	20.11	16.60	18.91	22.52	20.39	18.23	21.92	18.61	19.50	18.26	19.51
Average	80.76	80.62	81.88	80.22	83.03	81.13	83.74	81.78	80.57	81.43	81.52

Table 2: Summary of word accuracies for SPLICE using clean acoustic models

SPLICE	Set A				Set B				Set C		
	N1	N2	N3	N4	N1	N2	N3	N4	N1	N2	Average
CLEAN	99.48	99.40	99.37	99.48	99.48	99.40	99.37	99.48	99.57	99.49	99.45
SNR20	98.89	99.06	99.14	98.61	99.14	98.58	98.93	98.89	98.68	97.76	98.77
SNR15	97.64	98.55	98.45	97.78	98.53	97.25	98.06	97.62	97.18	95.56	97.66
SNR10	95.27	96.16	96.03	94.57	95.67	91.02	94.09	91.89	91.93	88.72	93.54
SNR5	87.96	81.80	82.64	83.74	83.39	67.74	77.57	71.03	75.04	68.20	77.91
SNR0	63.28	42.78	46.73	57.11	53.67	32.50	41.75	26.54	40.87	35.25	44.05
SNR-5	28.89	13.48	14.76	24.00	18.70	12.06	12.76	7.56	15.75	15.30	16.33
Average	88.61	83.67	84.60	86.36	86.08	77.42	82.08	77.19	80.74	77.10	82.39

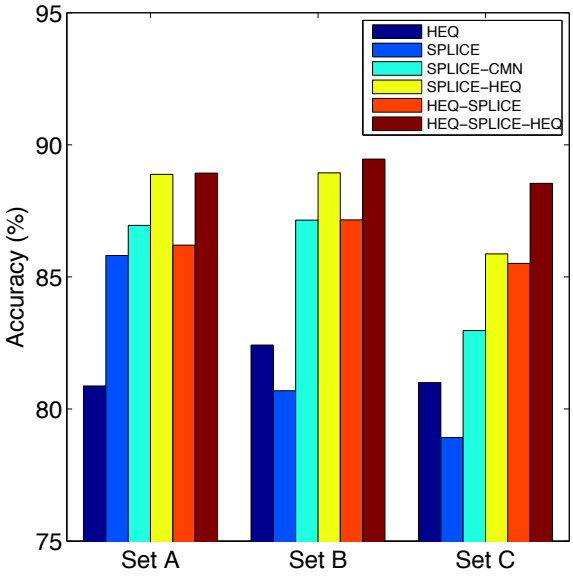

Figure 1: Average word accuracies of Aurora-2 recognition results using clean acoustic models

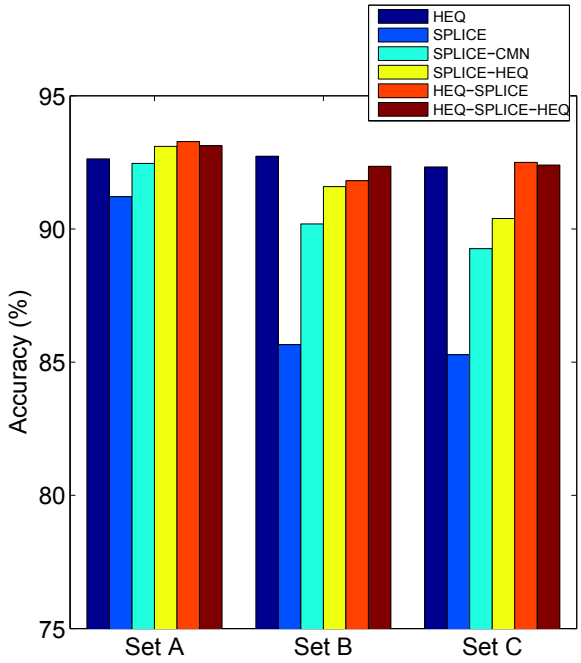

Figure 2: Average word accuracies of Aurora-2 recognition results using multi-conditions acoustic models

Table 3: Summary of word accuracies for HEQ-SPLICE-HEQ using clean acoustic models

HEQ-SPLICE-HEQ	Set A				Set B				Set C		
	N1	N2	N3	N4	N1	N2	N3	N4	N1	N2	Average
CLEAN	99.66	99.70	99.46	99.63	99.66	99.70	99.46	99.63	99.69	99.64	99.62
SNR20	98.71	99.00	99.02	98.89	98.96	98.61	99.11	99.04	98.50	98.40	98.82
SNR15	97.30	98.28	98.51	97.66	98.19	97.97	98.90	98.46	97.45	97.94	98.07
SNR10	95.12	96.04	96.48	94.69	96.19	95.28	97.23	95.83	95.27	94.92	95.71
SNR5	88.24	88.03	89.20	86.55	89.19	86.88	90.58	87.90	88.46	86.40	88.14
SNR0	66.56	60.13	61.29	68.93	67.15	60.85	69.79	63.13	66.29	61.79	64.59
SNR-5	29.94	22.82	21.47	36.62	29.94	25.03	29.91	26.29	29.54	25.03	27.66
Average	89.19	88.30	88.90	89.34	89.94	87.92	91.12	88.87	89.19	87.89	89.07

6 combinations of SPLICE and normalization: 1) SPLICE only, 2) HEQ only, 3) CMN after SPLICE, 4) HEQ after SPLICE, 5) HEQ before SPLICE, 6) HEQ both before and after SPLICE. GMM used in SPLICE has 1024 components and was trained in advance by using noisy speech data. Feature normalization was done utterance by utterance.

Figure 1 shows the averages of speech recognition results using clean acoustic models. Table 1, 2 and 3 show summaries of HEQ, SPLICE and HEQ-SPLICE-HEQ respectively. From Figure 1, we can say that SPLICE shows high performance in Set A but lower performance in Set B and Set C, and that HEQ shows balanced performance in contrast. The recognition accuracy is clearly improved by applying HEQ additionally to SPLICE (SPLICE-HEQ and HEQ-SPLICE). The performance of SPLICE-HEQ is better than SPLICE-CMN in all the test sets. By comparing SPLICE-HEQ and HEQ-SPLICE, it seems that HEQ should be introduced after SPLICE. But HEQ-SPLICE-HEQ shows the best result in all the test sets. Especially, the effectiveness of double HEQ is high in set C.

Figure 2 shows the averages of speech recognition results using multi-conditions acoustic models. On the whole, the similar tendency is observed as shown in Figure 1 but some differences are found. Compared with SPLICE-CMN, the accuracy of SPLICE-HEQ improves in all the sets. HEQ alone yields a high performance for each case. Therefore the features obtained by applying HEQ before SPLICE gave us a little improvement compared with HEQ only.

4. Conclusions

This paper describes effective combinations of SPLICE and feature normalization for noise robust speech recognition. In the experiments, the combination of HEQ and SPLICE yields higher performance in both clean condition and multi condition than the conventional combination. In clean condition, we achieved a 41% improvement in word error rate over SPLICE only, and a 25% improvement over the conventional combination of SPLICE and CMN.

Our future work is to compare our proposed method with another noise robust features like the advanced front-end [8]. Moreover, we plan to experiment on another database like Aurora-3 or Aurora-4 database.

References

[1] C. R. Jankowski Jr., H.-D. H. Vo, and R. P. Lippmann, "A comparison of signal processing front ends for automatic word recognition," *IEEE Trans. on Speech and Audio Processing*, vol. 3, no. 4, pp. 286–293, 1995.

[2] O. Viikki and K. Laurila, "Cepstral domain segmental feature vector normalization for noise robust speech recognition," *Speech Communication*, vol. 25, pp. 133–147, 1998.

[3] A. de la Torre, A. M. Peinado, J. C. Segura, J. L. Perez-Cordoba, and M. C. Benitez, and A. J. Rubio, "Histogram Equalization of Speech Representation for Robust Speech Recognition," *IEEE Trans. on Speech and Audio Processing*, vol. 13, no. 3, pp. 355–366, 2005.

[4] Y. Suh, M. Ji., and H. Kim, "Probabilistic Class Histogram Equalization for Robust Speech Recognition," *IEEE Signal Processing Letters*, vol. 14, no. 4, pp. 287–290, 2007.

[5] J. Droppo, L. Deng and A. Acero, "Evaluation of the SPLICE on the Aurora2 and 3 Tasks," *Proc. ICSLP*, pp. 29–32, 2002.

[6] M. Suzuki, T. Yoshioka, S. Watanabe, N. Minematsu, and K. Hirose, "Framewise MFCC Enhancement in Observation and Noise Feature Space," *Proc. ICASSP*, 2012.

[7] H. G. Hirsch and D. Pearce, "The AURORA experimental framework for the performance evaluation of speech recognition systems under noisy conditions," *Proc. ISCA ITRW ASR2000*, pp. 181–188, 2000.

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