Online Blind Channel Normalization Using BPF-Based Modulation Frequency Filtering

Yun-Kyung Lee, Ho-Young Jung, and Jeon Gue Park

We propose a new bandpass filter (BPF)-based online channel normalization method to dynamically suppress channel distortion when the speech and channel noise components are unknown. In this method, an adaptive modulation frequency filter is used to perform channel normalization, whereas conventional modulation filtering methods apply the same filter form to each utterance. In this paper, we only normalize the two mel frequency cepstral coefficients (C0 and C1) with large dynamic ranges; the computational complexity is thus decreased, and channel normalization accuracy is improved. Additionally, to update the filter weights dynamically, we normalize the learning rates using the dimensional power of each frame. Our speech recognition experiments using the proposed BPF-based blind channel normalization method show that this approach effectively removes channel distortion and results in only a minor decline in accuracy when online channel normalization processing is used instead of batch processing.

Keywords: Channel normalization, Speech recognition, Adaptive filter modeling, Modulation frequency filtering.

I. Introduction

With the recent increase in the use of speech recognition technologies in various speech communication services, efficient channel normalization and noise reduction, have become important for enhancing speech quality and improving speech recognition accuracy [1]–[5]. In general, the previous methods for channel normalization and noise reduction use identical filters for each speech signal channel, and perform normalization and noise reduction on the entire input speech signal after sentences have been completed. This contributes to undesired discontinuities under realistic speech recognition conditions [6]–[9].

To solve this problem, we propose an online channel normalization method to model a bandpass filter (BPF)-based adaptive filter and calculate the filter coefficients for each channel. High-pass filter (HPF)-based adaptive filters efficiently reduce the slow-varying noise components in the feature domain. However, they tend to emphasize the fast-varying noise components. We calculate the channel normalization filter by applying a low-pass filter (LPF) to the HPF-based adaptive filter, and perform channel normalization only on the C0 and C1 components of the mel frequency cepstral coefficient (MFCC) feature vector sequence, to decrease the computational complexity in real environments. In addition, the proposed method dynamically adjusts the learning rates to reduce convergence time and improve the feature-extraction accuracy; in contrast, the previous channel normalization methods use a fixed learning rate when calculating the filter coefficients. The speech recognition results obtained using a mobile-voice search database show that the proposed method has almost no performance degradation under online speech recognition setups compared to batch

Manuscript received Nov. 18, 2015; revised Aug. 8, 2016; accepted Aug. 25, 2016.

This work was supported by the ICT R&D program of MSIP/IITP (R0126-15-1117, Core technology development of the spontaneous speech dialogue processing for the language learning).

Yun-Kyung Lee (corresponding author, yunklee@etri.re.kr), Ho-Young Jung (hjung@etri.re.kr), and Jeon Gue Park (jgp@etri.re.kr) are with the SW & Content Research Laboratory, ETRI, Daejeon, Rep. of Korea.

channel normalization results.

The remainder of this paper is organized as follows. Section II describes the signal model, the dynamic learning rules used to calculate the filter weights, and the proposed BPF-based blind channel normalization filter approach. Section III describes the experimental results, and Section IV offers some concluding remarks.

II. BPF-Based Blind Channel Normalization Filter

1. Signal Modeling

In this paper, channel normalization was conducted using a BPF-based adaptive filter. Channel distortion and additive channel noise are predominantly slow-varying perturbations, which causes temporal dependencies in the feature vector domain. To statistically remove these dependencies and perform blind channel normalization, we use an information-maximization approach that maximizes the joint entropy in the feature vector domain. The information-maximization approach is modeled simply as a finite impulse response - formed unsupervised adaptive HPF in the modulation frequency space [10], [11]. However, conventional HPF-based normalization filters have a feature vector discontinuity problem between adjacent normalized frames, and tend to emphasize the fast-varying noise components. To overcome these problems, we used a BPF-based filter to conduct channel normalization, by applying an LPF to the modeled HPF-based normalization filter. Figure 1 shows a schematic diagram of blind channel normalization based on such an adaptive filtering approach [12], [13].

In Fig. 1, Y denotes the distorted input feature vector sequence, U the normalized feature vector sequence at the output of the BPF-based adaptive filter W, $g(\cdot)$ the activation function used to train the filter weights, and X the output frame feature.

The filtered feature vector $U(t)$ and output feature vector $X(t)$ are defined as follows:

$$U(t) = \sum_{j=0}^{J}\sum_{k=0}^{K} w_j^{\mathrm{L}} \cdot w_k^{\mathrm{H}} \cdot Y(t-j-k), \tag{1}$$

$$X(t) = g(U(t)), \tag{2}$$

where w_j^{L} and J respectively denote the jth coefficient and the order of the low-pass filter W^{L}, w_k^{H} and K denote the kth coefficient and order of the high-pass filter W^{H}, and t denotes the frame index.

Given that multiplication in the frequency domain is equivalent to convolution in the time domain, the BPF-based channel normalization can also be computed by applying a smoothing process to the high-pass filtered output feature vector sequences [14]. After HPF-based filtering, the distorted input feature vector $U^{\mathrm{H}}(t)$ and output feature vector $X^{\mathrm{H}}(t)$ are represented as

$$U^{\mathrm{H}}(t) = \sum_{k=0}^{K} w_k^{\mathrm{H}} \cdot Y(t-k), \tag{3}$$

$$X^{\mathrm{H}}(t) = g\left(U^{\mathrm{H}}(t)\right). \tag{4}$$

The frequency response of the low-pass filter W^{L} is defined as

$$F(Z) = 1 + \alpha \cdot Z^{-1}. \tag{5}$$

Therefore, the smoothed output feature vector $\tilde{X}(t)$ can be computed in the time domain as follows:

$$\tilde{X}(t) = X^{\mathrm{H}}(t) + \alpha \cdot X^{\mathrm{H}}(t-1). \tag{6}$$

In this paper, $\alpha = 0.98$ is used for smoothing; the final output feature vector is therefore defined as

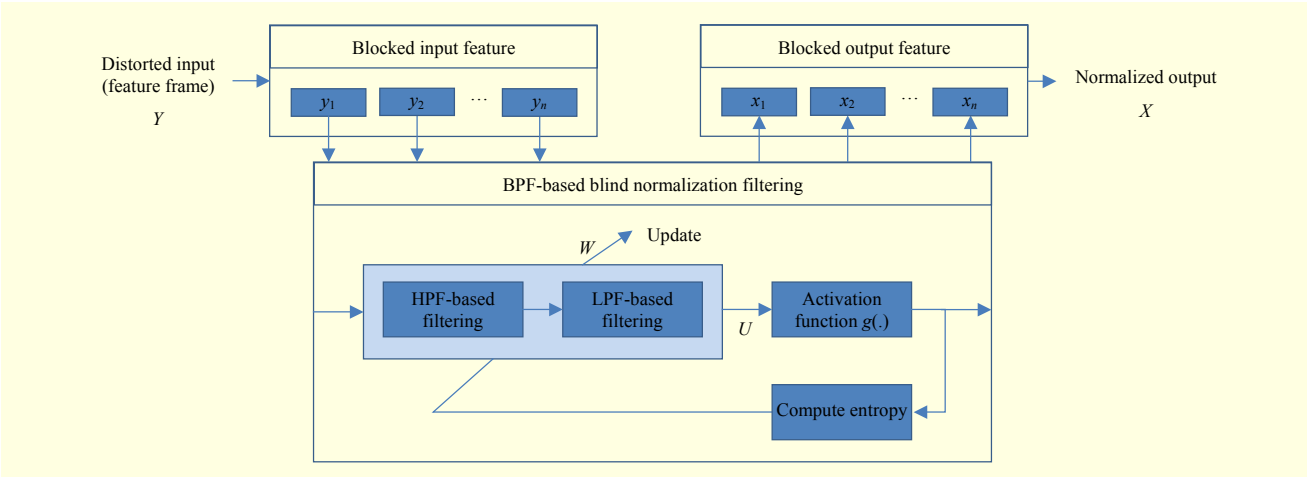

Fig. 1. Block diagram of the BPF-based blind channel normalization filter.

$$\tilde{X}(t) = X^{\mathrm{H}}(t) + 0.98 \cdot X^{\mathrm{H}}(t-1). \quad (7)$$

2. Dynamic Learning Rule for the Filter Weights

The learning rates used to train the filter weights have a major impact on the maximization of the joint entropy of the feature vectors. Depending on the learning rates, the filter coefficients can both diverge or converge to local maxima, which degrades channel normalization or speech recognition. In this paper, we normalized the filter coefficients learning rates, using the dimensional power of each feature vector in the filter weight update process; this has the same effect of using a dynamic learning rate that changes according to the gradient of each utterance and channel.

To apply the information-maximization theory, the joint entropy $H(\tilde{X})$ is defined as in [7]:

$$H(\tilde{X}) = -\mathrm{E}\left[\ln f\left(\tilde{X}(t)\right)\right], \quad (8)$$

where $\mathrm{E}[\cdot]$ denotes the expectation operator, and $f(\tilde{X})$ is the probability density function (PDF) of the output feature vector sequence $\tilde{X}$, given by

$$f(\tilde{X}) = \frac{f(Y)}{\left|\frac{\partial \tilde{X}}{\partial Y}\right|}. \quad (9)$$

The joint entropy $H(\tilde{X})$ defined in (8) can be expanded as

$$\begin{aligned} H(\tilde{X}) &= -\mathrm{E}\left[\ln f(Y(t)) - \ln\left|\frac{\partial \tilde{X}(t)}{\partial Y(t)}\right|\right] \\ &= \mathrm{E}\left[\ln\left|\frac{\partial \tilde{X}(t)}{\partial Y(t)}\right|\right] - \mathrm{E}\left[\ln f\left(Y(t)\right)\right]. \end{aligned} \quad (10)$$

To maximize the joint entropy with respect to the filter coefficients w_k^{H} only the first term in (10) needs to be considered, because the second term is not affected by changes in w_k^{H}.

The gradient descent rule for w_k^{H} is computed by taking the gradient of that first term, and is defined as

$$\begin{aligned} \Delta w_k^{\mathrm{H}} &= \mathrm{E}\left[\frac{\partial \ln\left|\tilde{X}'(t)\right|}{\partial w_k^{\mathrm{H}}}\right] \\ &= \mathrm{E}\left[\frac{1}{\tilde{X}'(t)}\frac{\partial \tilde{X}'(t)}{\partial w_k^{\mathrm{H}}}\right], \end{aligned} \quad (11)$$

where $\tilde{X}'(t)$ can be expanded as

$$\begin{aligned} \tilde{X}'(t) &= \frac{\partial \tilde{X}(t)}{\partial Y(t)} \\ &= \frac{\partial \tilde{X}(t)\partial U(t)}{\partial U(t)\partial Y(t)} = g'\left(U(t)\right)\cdot w_0^{\mathrm{H}}. \end{aligned} \quad (12)$$

Therefore, $\partial \tilde{X}'(t)/\partial w_k^{\mathrm{H}}$ in (11) can be computed as:

$$\frac{\partial \tilde{X}'(t)}{\partial w_k^{\mathrm{H}}} = g'\left(U(t)\right)\cdot\frac{\partial w_0^{\mathrm{H}}}{\partial w_k^{\mathrm{H}}} + \frac{\partial g'\left(U(t)\right)}{\partial w_k^{\mathrm{H}}}\cdot w_0^{\mathrm{H}}. \quad (13)$$

The activation function $g(\cdot)$ is used to update the filter weights, and can be assumed to be a sigmoid, a Gaussian distribution, or some other appropriate function. In this paper, we used the Gaussian distribution given in [15]:

$$g'\left(U(t)\right) = Ce^{-U^2(t)}, \quad (14)$$

$$\frac{\partial g'\left(U(t)\right)}{\partial w_k^{\mathrm{H}}} = g'\left(U(t)\right)\cdot\left[-2U(t)Y(t-k)\right]. \quad (15)$$

After obtaining the learning rules for w_k^{H} by combining (14), (15), and (11), we normalized them by dividing each feature vector by their dimensional power, before dynamically updating the filter weights. The learning rules for w_k^{H} used in this paper are therefore defined as

$$\Delta w_k^{\mathrm{H}} = \begin{cases} \mathrm{E}\left[\dfrac{1}{w_0^{\mathrm{H}}} - 2\dfrac{U(t)}{\|U(t)\|}\dfrac{Y(t)}{\|Y(t)\|}\right], & k = 0, \\ \mathrm{E}\left[-2\dfrac{U(t)}{\|U(t)\|}\right], & \text{otherwise.} \end{cases} \quad (16)$$

The filter coefficients w_k^{H} are iteratively updated by

$$w_{k,i+1}^{\mathrm{H}} = w_{k,i}^{\mathrm{H}} + \eta\Delta w_k^{\mathrm{H}}, \quad (17)$$

where i denotes the iteration index, and η denotes the learning rate used to update the filter coefficients.

3. BPF-Based Blind Channel Normalization Filter

In a real environment, we cannot know the original speech signal or channel noise component. In addition, channel normalization systems must work in real-time. For this reason, we conducted online blind channel normalization using the BPF-based normalization filter and dynamic learning rates—to update the filter weights—discussed above.

The proposed BPF-based channel normalization scheme proceeds as follows:

(S1) Initialize the filter coefficients w_k^{H} and the sequences $U(t)$ and $X^{\mathrm{H}}(t)$, using (3) and (4).
(S2) Compute the gradient descent rule for w_k^{H} using (11).
(S3) Normalize and update the filter coefficients with (16) and (17), and calculate the new sequences $U(t)$ and $X^{\mathrm{H}}(t)$.
(S4) Apply the smoothing process using (7).
(S5) Iterate (S2), (S3), and (S4) until the convergence criterion for the filter coefficients is met. In this paper, we used a threshold of 0.00001 as stopping criterion.
(S6) Extract the output feature vector sequence to remove channel noise, and normalize the feature vector using (4) and (7).

III. Experimental Results and Discussion

1. Speech Database

We used the mobile-voice search (MVS) database, which was gathered from commercial mobile service and contains various users under realistic voice search conditions, in the street, bus, metro, office, and home environments. The database consists of two subsets: a distorted dataset gathered in December (*Dec. noisy*), and datasets gathered in August (*Aug. normal* and *Aug. noisy*). The December and August datasets have different MVS system users and different environments. In the August dataset, the speech signals were manually (humanly) tagged and divided into two groups, to compare the performance difference between noisy and normal conditions, whereas the December dataset used all speech signals in one group. In the *Aug. normal* dataset, the speech signals were collected with stationary background noise or in quiet environments.

The sampling rate of the speech database used in this study was 16 kHz. The feature vectors were computed on 20-ms speech segments, with an overlap of 10 ms between adjacent frames. For each frame, 23 mel-scaled filterbank energies were derived, normalized by their frame energy, and scaled logarithmically. After filtering with the proposed blind normalization filtering approach, 13 MFCCs were extracted by taking a discrete cosine transform. We then derived 39 dynamic feature vectors (inter-frame features) and one intra-log energy measure from the 13 MFCC features [2]. For our speech recognition experiments, we used 53 feature vector sequences (13 MFCCs + 39 dynamic features + 1 intra-log energy).

In the proposed channel normalization process, the static features (13 MFCCs: C0 through C12) were normalized; the 39 dynamic features have inherently time-normalized characteristics. In general, the C0 and C1 components of the MFCC features have a large variance, whereas components C2 to C12 have insignificant variance values. Hence, the normalized values of the C2 to C12 components do not differ much from their original values. Only C0 and C1 were therefore normalized, which is an efficient way of decreasing the computational complexity in real environments. Figure 2 shows an example of the variance of the static components (C0 to C12).

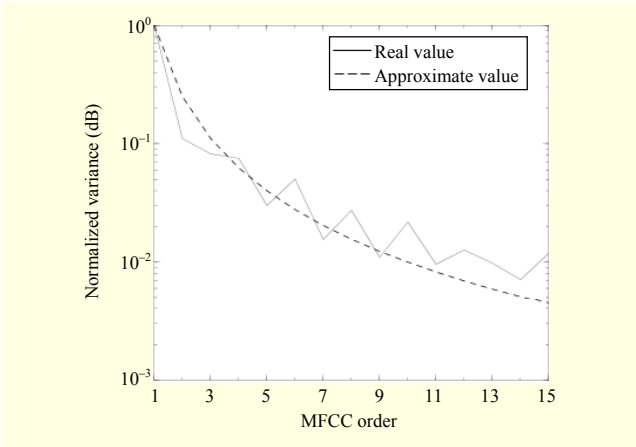

Fig. 2. Example of the variance of the MFCCs components.

The learning rate η was 0.001 for C0, and 0.0001 for C1. The threshold for establishing convergence was 0.00001, and a filter order of 10 was chosen in this paper. The learning rates and threshold were determined experimentally.

2. Results of Channel Normalization

To validate the performance of the channel normalization scheme, we compared the plots of C0 and C1 of the input feature vector sequence and those of the normalized feature vector sequence obtained with the proposed approach, under batch and block online processing conditions.

The speech recognition accuracy and error reduction rate (ERR) were also computed, to evaluate performance quantitatively. One of the conventional equivalent average filter-based channel normalization methods, cepstral mean subtraction (CMS) [3], was used as an ERR reference for performance comparison.

A. Waveform and Feature Vector Sequence Plot

Figures 3 and 4 show some examples of input waveforms and the corresponding feature vector sequence plots. Figure 3 shows an input speech signal waveform, the plot of the corresponding input C0 feature vector sequence, and the normalized feature vector sequences obtained after filtering with a batch normalization filter and a block online normalization filter. Figure 4 shows plots of the input C1 feature vector sequence, the batch-filtered feature vector sequence, and the block online-filtered feature vector sequence.

As mentioned above, we only used C0 and C1 (which have a large dynamic range), to reduce computational complexity and improve the normalization performance. Comparing the C0 and C1 plots, we confirmed that the feature vector sequences were biased efficiently, yielding channel-normalized feature vector sequences. Furthermore, the block online results have almost the same shape as the batch channel normalization results.

B. Speech Recognition Results

Tables 1 and 2 show the speech recognition results obtained in the batch and block online experimental setups. As shown, the proposed BPF-based blind channel normalization filtering approach effectively removes channel distortion, and does so

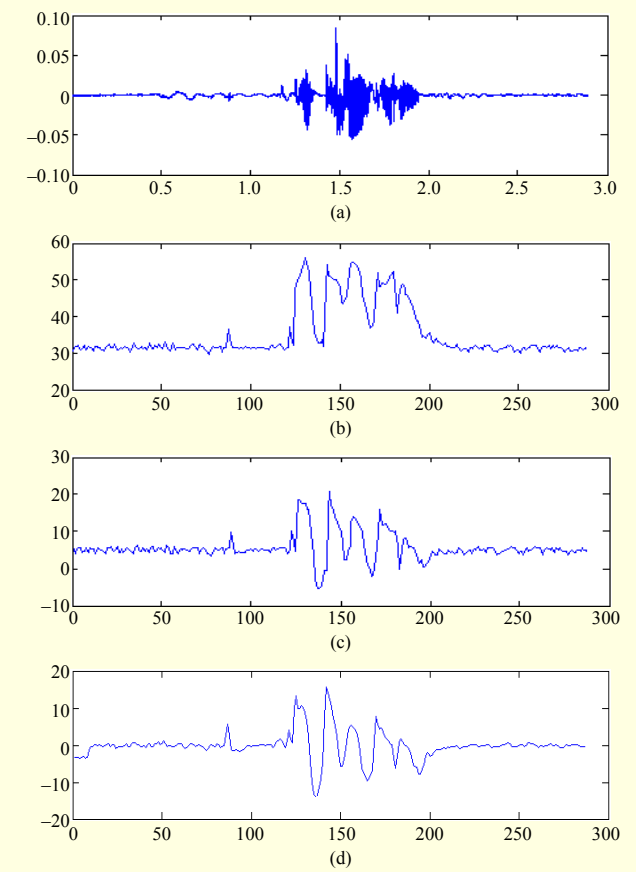

Fig. 3. Speech waveform and C0 feature vector sequences. (a) Input speech signal waveform. (b) Input signal feature vector sequence. (c)–(d) Output feature vector sequences using (c) batch and (d) block online normalization filters.

better than both the previous HPF-based and baseline methods. Additionally, we confirmed that, compared to the batch results (which used all static features to normalize the channel), the proposed approach maintained the same system performance in real-time (block online) setups using only C0 and C1.

We also calculated the ERR of the speech recognition results, which can be defined as

$$ERR\,(\%) = \frac{Acc.^{N} - Acc.^{B}}{error} \cdot 100, \qquad (18)$$

where $Acc.^{N}$ and $Acc.^{B}$ represent the speech recognition accuracy after and before filtering, respectively, and *error* represents the speech recognition error.

Figure 5 shows the ERR scores for the speech recognition results obtained using both the CMS and the proposed BPF-based filtering approaches. Overall, the proposed method exhibits an almost identical performance for both batch and block online conditions. In addition, the proposed method reduces the performance degradation resulting from applying the system in real-time setup compared to the conventional CMS approach.

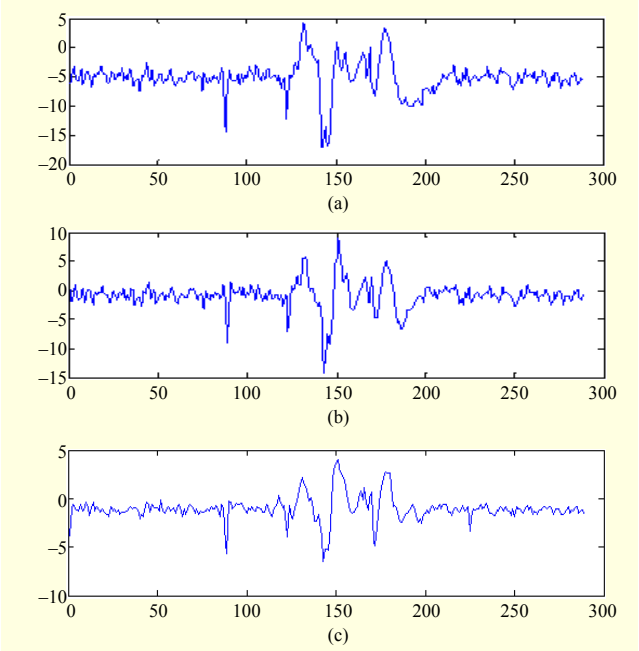

Fig. 4. C1 feature vector sequences. (a) Input signal feature vector sequence. (b)–(c) Output feature vector sequences using (b) batch and (c) block online normalization filters.

Table 1. Speech recognition results (%) of previous methods.

		Baseline method (MFCC)	Previous HPF-based method
Dec.		47.1	58.12
Aug.	Normal	72.1	80.25
	Noisy	39.0	61.10

Table 2. Speech recognition results (%) of the proposed channel normalization filtering approach.

Proposed method		All MFCCs		C0 and C1	
		Batch	Block online	Batch	Block online
Dec.		62.52	61.85	62.82	62.39
Aug.	Normal	84.75	83.58	84.82	84.42
	Noisy	65.09	62.58	65.89	64.48

IV. Conclusion

We proposed a new BPF-based blind channel normalization filtering approach, capable of removing the channel distortion and suppressing channel noise in real environments. In the proposed approach, the normalization filter is modeled as a

ETRI Journal, Volume 38, Number 6, December 2016
http://dx.doi.org/10.4218/etrij.16.0115.0994

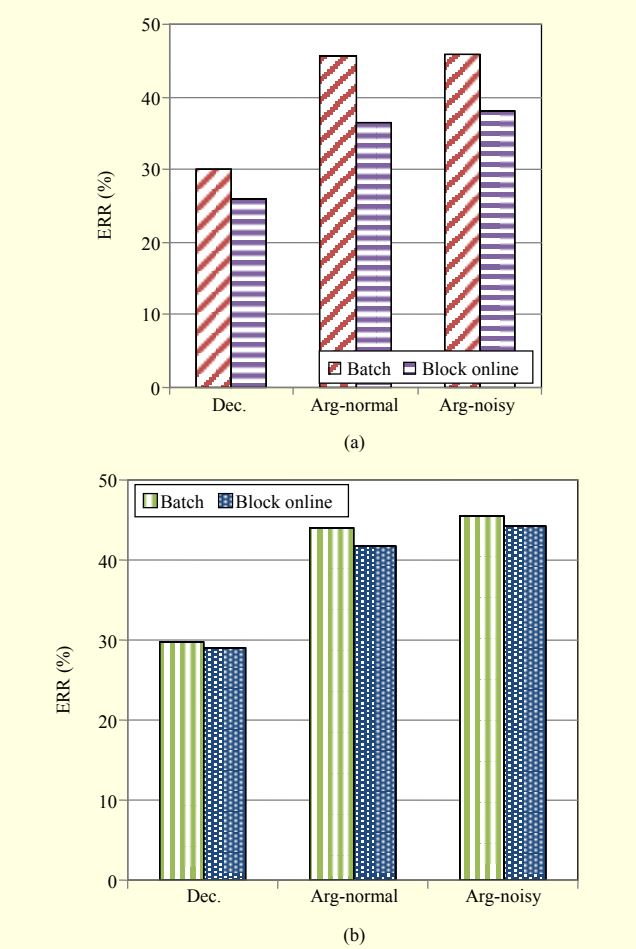

Fig. 5. ERR scores for the speech recognition results obtained with (a) CMS and (b) proposed approach.

BPF, because HPF-based adaptive filtering results have sparsity and discontinuity problems between adjacent frames. The proposed approach iteratively updates the filter coefficients by adopting the gradient descent rule. Because the learning rate is dependent on the range of changes in the learning rules, we updated the filter coefficients dynamically, using the dimensional power of each feature vector sequence. To decrease computational complexity, only the C0 and C1 elements of the MFCC feature vector were used in this paper.

We showed that signal normalization removed channel distortion by providing the plots of the normalized feature vector sequences. Through speech recognition tests, we also confirmed that the proposed approach was capable of maintaining the speech recognition accuracy of the batch condition, even under block online conditions. In fact, the ERR scores obtained from the speech recognition results show a similar system performance for both batch and block online setups. The experimental results confirmed that the proposed BPF-based adaptive filtering approach is useful for online blind channel normalization systems.

References

[1] H.J. Song, Y.K. Lee, and H.S. Kim, "Probabilistic Bilinear Transformation Space-Based Joint Maximum a Posteriori Adaptation," *ETRI J.*, vol. 34, no. 5, Oct. 2010, pp. 783–786.

[2] S.J. Lee et al., "Intra-and Inter-frame Features for Automatic Speech Recognition," *ETRI J.*, vol. 36, no. 3, June 2014, pp. 514–517.

[3] H.-Y. Jung, "On-line Blind Channel Normalization for Noise-Robust Speech Recognition," *IEIE Trans. Smart Process. Comput.*, vol. 1, no. 3, Dec. 2012, pp. 143–151.

[4] Y. Ephraim and D. Malah, "Speech Enhancement Using a Minimum Mean-Square Error Short-Time Spectral Amplitude Estimator," *IEEE Trans. Acoustics, Speech Signal Process.*, vol. 32, no. 6, Dec. 1984, pp. 1109–1121.

[5] S. Sigurdsson, K.B. Petersen, and T. Lehn-Schiøle, "Mel Frequency Cepstral Coefficients: an Evaluation of Robustness of mp3 Encoded Music," *Proc. Int. Conf. Music Inform. Retrieval*, Victoria, Canada, Oct. 8–12, 2006.

[6] M.M. Rahman et al., "Performance Evaluation of CMN for Mel-LPC based Speech Recognition in Different Noisy Environments," *Int. J. Comput. Appl.*, vol. 58, no. 10, 2012, pp. 6–10.

[7] H. Hermansky and N. Morgan, "RASTA Processing of Speech," *IEEE Trans. Speech Audio Process.*, vol. 2, no. 4, Oct.1994, pp. 578–589.

[8] H. You and A. Alwan, "Temporal Modulation Processing of Speech Signals for Noise Robust ASR," *Auun. Conf. Int. Speech Commun. Associateion*, Brighton, UK, Sept. 6–10, 2009, pp. 36–39.

[9] J.A. Cadzow, "Blind Deconvolution via Cumulant Extrema," *IEEE Signal Process. Mag.*, vol. 13, no. 3, May 1993, pp. 24–42.

[10] A.J. Bell and T.J. Sejnowski, "An Information-Maximization Approach to Blind Separation and Blind Deconvolution," *Neural Comput.*, vol. 7, no. 6, Apr. 1995, pp. 1129–1159.

[11] H.H. Yang and S. Amari, "Adaptive On-line Learning Algorithms for Blind Separation – Maximum Entropy and Minimum Mutual Information," *Neural Comput.*, vol. 9, no. 7, 1997, pp. 1457–1482.

[12] P.C. Loizou, *Speech enhancement*, Boca Raton, FL, USA: CRC Press, 2007, pp. 97–289.

[13] Papoulis, *Probability, Random Variables, and Stochastic Processes*, Chicago IL, USA: McGraw-Hill, 1991.

[14] A.V. Oppenheim and R.W. Schaefer, *Digital signal processing*, Upper Saddle River, NJ, USA: Prentice-Hall, 1989.

[15] H. Shen, G. Liu, and J. Guo, "Two-Stage Model-based Feature Compensation for Robust Speech Recognition," *Comput.*, vol. 94, no. 1, 2012, pp. 1–20.

Yun-Kyung Lee received the BS degree in Electronics Engineering and the MS degree in Control and Instrumentation Engineering from Chungbuk National University (CBNU), Cheongju, Rep. of Korea, in 2007 and 2009, respectively. She received the PhD degree in Control and Robot Engineering at CBNU, in 2013. She is now in charge of the Spoken Language Processing Research Section, ETRI, Daejeon, Rep. of Korea. Her research interests are speech processing and automatic speech recognition technology.

Ho-Young Jung received the MS and PhD degrees in Electrical Engineering from the Korea Advanced Institute of Science and Technology, Daejeon, Rep. of Korea, in 1995 and 1999, respectively. His PhD dissertation focused on robust speech recognition. He joined ETRI, in 1999 as a senior researcher, and has belonged to the automatic translation and language intelligence research department as a principal researcher. His current research interests include noisy speech recognition, spontaneous speech understanding, machine learning, and cognitive computing. He has published or presented more than 35 papers in the field of spoken-language processing.

Jeon Gue Park received his PhD degree in Information and Communication Engineering from Paichai University, Daejeon, Rep. of Korea, in 2010. He is currently in charge of the Spoken Language Processing Research Section, ETRI. His current research interests include speech recognition and dialogue systems, artificial intelligence, and cognitive systems.

ETRI Journal, Volume 38, Number 6, December 2016
http://dx.doi.org/10.4218/etrij.16.0115.0994