SEQUENTIAL NON-STATIONARY NOISE TRACKING USING PARTICLE FILTERING WITH SWITCHING DYNAMICAL SYSTEM

Masakiyo Fujimoto and Satoshi Nakamura

ATR Spoken Language Communication Research Laboratories
2-2-2, Hikari-dai, Seika-cho, Souraku-gun, Kyoto, 619-0288, Japan
E-mail: {masakiyo.fujimoto, satoshi.nakamura}@atr.jp

ABSTRACT

This paper addresses a speech recognition problem in non-stationary noise environments: the estimation of noise sequences. To solve this problem, we present a particle filter-based sequential noise estimation method for the front-end processing of speech recognition. In the proposed method, the particle filter is defined by a dynamical system based on Polyak averaging and feedback. We also introduce a switching dynamical system into the particle filter to cope with the state transition characteristics of non-stationary noise. In the evaluation results, we observed that the proposed method improves speech recognition accuracy in the results of non-stationary noise environments by a noise compensation method with stationary noise assumptions.

1. INTRODUCTION

Noise robustness is one of the most important problems for the application of speech recognition techniques in real environments. For this problem a lot of research in speech recognition in noise has been reported where adverse noise is restricted to stationary noise [1, 2, 3]. However, most of the noise observed in real environments has non-stationary characteristics. To improve speech recognition accuracy in non-stationary noise environments, it is necessary to estimate the noise sequence as accurately as possible. However, the estimation of non-stationary noise sequences is difficult because, in most cases, the observable signal on which the speech recognition is performed is the only noise added speech signal. So both clean speech and noise have non-stationary characteristics.

To solve such problems, several estimation methods of non-stationary noise based on a sequential EM algorithm are reported [4, 5, 6] that can effectively estimate noise sequences. However, their computation costs are expensive because frame by frame iterative estimation is required for the convergence of noise parameters. Recently, a particle filter-based sequential estimation [7, 8] has attracted attention and been applied to various research fields. The particle filter is a Bayesian estimation method, whose main estimation framework is based on a sequential Monte Carlo method. Therefore, the computation costs of the particle filter are cheaper than a sequential EM algorithm because iterative estimation is not always required.

In this paper, we present a sequential non-stationary noise estimation method based on particle filtering. In applications of particle filtering, first, a definition of the signal model called a dynamical system (state-space model) is required. Typically, a dynamical system can be defined by two equations: a state transition equation that represents the dynamics of the target signal, and an observation equation that represents the output system of the observed signal. In our previous work [9], we proposed a sequential non-stationary noise estimation method with a Polyak averaging-based state transition equation [6, 10]. The Polyak averaging-based state transition equation has three parameters and these are set as time constant parameters in our previous work. However, these parameters should be set as time varying parameters because the state transition characteristics of non-stationary noise may change frame by frame. In this problem, we introduce a switching dynamical system [11], which changes the parameters of Polyak averaging, into particle filter-based sequential estimation and show its effectiveness in noise estimation and the improvement of speech recognition accuracy.

Our proposed method was evaluated on a connected Japanese digit recognition task [12] conducted on speech recognition in highly non-stationary noise environments. In the evaluation results, we observed that the proposed method improves speech recognition accuracy in the results of non-stationary noise environments by a noise compensation method with stationary noise assumptions.

2. PARTICLE FILTER-BASED NOISE ESTIMATION

Figure 1 is a block diagram of the proposed front-end environmental compensation method that consists of particle filter-based parameter estimation and a minimum mean square error (MMSE)-based clean speech estimation method [3]. In the particle filter, the parameters are estimated through sequential importance sampling that consists of extended Kalman filter-based parameter updating, a sample weight computation, residual resampling [7], and a Markov chain Monte Carlo with Metropolis-Hastings sampling [13]. The details of the proposed method are described below.

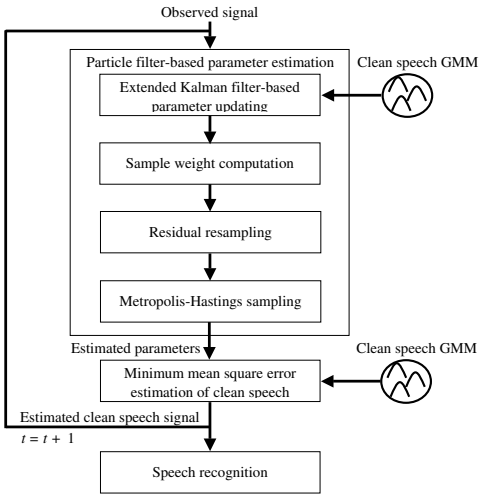

Fig. 1. A block diagram of the proposed front-end environmental compensation method

2.1. Particle filtering algorithm

Sequential importance sampling step

Let $\mathbf{X}_t$ and $\mathbf{N}_t$ denote the vectors at t-th short time frame that have logarithmic output energy of Mel-filter bank of observed noisy speech and noise, respectively. When $\mathbf{X}_t$ is observed, the *a posteriori* probability density function (PDF) for the sequence of $\mathbf{N}_t$ can be represented by Eq. (1) under the assumption that the sequence of the *a posteriori* PDF of $\mathbf{N}_t$ follows a first order Markov chain.

$$p(\mathbf{N}_{0:t}|\mathbf{X}_{0:t}) = p(\mathbf{N}_0|\mathbf{X}_0) \prod_{t'=1}^{t} p(\mathbf{N}_{t'}|\mathbf{N}_{t'-1}) p(\mathbf{X}_{t'}|\mathbf{N}_{t'}) \ , \quad (1)$$

where $\mathbf{N}_{0:t} = \{\mathbf{N}_0, \ldots, \mathbf{N}_t\}$ and $\mathbf{X}_{0:t} = \{\mathbf{X}_0, \ldots, \mathbf{X}_t\}$. Therefore, $\mathbf{N}_{0:t}$ is estimated as the signal that recursively maximizes the above PDF. In a particle filtering, *a posteriori* PDF $p(\mathbf{N}_{0:t}|\mathbf{X}_{0:t})$ is approximated by Monte Carlo sampling as follows:

$$\begin{aligned} p(\mathbf{N}_{0:t}|\mathbf{X}_{0:t}) &\simeq \frac{1}{J} \sum_{j=1}^{J} \delta\left(\mathbf{N}_{0:t} - \mathbf{N}_{0:t}^{(j)}\right) \\ &\simeq \sum_{j=1}^{J} w_t^{(j)} p\left(\mathbf{N}_{0:t}^{(j)}|\mathbf{X}_{0:t}\right) \ , \end{aligned} \quad (2)$$

where j, J, $w_t^{(j)}$, and $\delta(\cdot)$ denote the sample index, the number of samples, the weight of sample j at time t ($\sum_{j=1}^{J} w_t^{(j)} = 1$), and a Dirac delta function, respectively.

Usually, samples $\mathbf{N}_t^{(j)}$ were drawn by a sequential importance sampling (SIS) [7] and sample weight $w_t^{(j)}$ is defined as

$$w_t^{(j)} \propto w_{t-1}^{(j)} p\left(\mathbf{X}_t|\mathbf{N}_t^{(j)}\right) \ . \quad (3)$$

In Eq. (2), $p(\mathbf{N}_{0:t}^{(j)}|\mathbf{X}_{0:t})$, *a posteriori* PDF for each sample, is updated from the previous PDF $p(\mathbf{N}_{0:t-1}^{(j)}|\mathbf{X}_{0:t-1})$ by using an extended Kalman filter.

Residual resampling (selection) step

In practice, after the SIS step, the weights of all but several samples may become insignificant. Given the fixed number of samples, this will degenerate the estimation. A selection step by residual resampling [7] is adopted after the sampling step. The method avoids degeneracy by discarding samples with insignificant weights, and to maintain a constant number of samples, those with significant weights are duplicated. Accordingly, weights after the selection step are also proportionally redistributed.

Markov chain Monte Carlo step

After the resampling step at frame t, these J samples are distributed approximately according to Eq. (2). However, the discrete nature of the approximation can skew the importance weight distribution, where in extreme cases all samples have the same value. A Metropolis-Hastings (MH) sampling [13] step is introduced in each sample that involves sampling a candidate given the current state according to a proposed *importance distribution*. To simplify the calculation, we assume that the importance distribution is symmetric.

2.2. Dynamical system based on Polyak averaging and feedback

In the SIS step, to update the *a posteriori* PDF for each sample using an extended Kalman filter, a definition of the signal model called a dynamical system (state-space model) is required. Typically, a dynamical system can be defined by two equations: a state transition equation that represents the dynamics of the target signal, and an observation equation that represents the output system of the observed signal.

First, assume that clean speech $\mathbf{S}_t$ can be modeled by a Gaussian mixture model (GMM). At time t, parameter $\mathbf{S}_{k_t,t}^{(j)}$ is sampled from Gaussian distribution k_t contained in a clean speech GMM. In this case, the observation process of $\mathbf{X}_t$ can be modeled by the following equation by using noise sample $\mathbf{N}_t^{(j)}$ and error signal $\mathbf{V}_t^{(j)}$,

$$\mathbf{X}_t = \mathbf{S}_{k_t,t}^{(j)} + \log\left(\mathbf{I} + \exp\left(\mathbf{N}_t^{(j)} - \mathbf{S}_{k_t,t}^{(j)}\right)\right) + \mathbf{V}_t^{(j)} \quad (4)$$

$$\mathbf{V}_t^{(j)} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Sigma}_{\mathbf{S}^{(j)},k_t}\right) \ , \quad (5)$$

where $\mathbf{\Sigma}_{\mathbf{S}^{(j)},k_t}$ denotes the diagonal covariance matrix of Gaussian distribution contained in clean speech GMM.

On the other hand, the state transition process of $\mathbf{N}_t^{(j)}$ is typically modeled by a random walk process as follows:

$$\mathbf{N}_{t+1}^{(j)} = \mathbf{N}_t^{(j)} + \mathbf{W}_t^{(j)} \ , \quad (6)$$

where $\mathbf{W}_t^{(j)}$ denotes the driving noise that follows zero mean Gaussian noise with diagonal covariance matrix of $\mathbf{\Sigma}_{\mathbf{W}}^{(j)}$

The random walk process is widely used for the state transition process of the dynamical system. However, it cannot represent the accurate state transition of noise because it provides the state transition of the target signal using random noise. Generally, the definition of the state transition equation is the most important factor for the state-space model-based accurate estimation of a noise sequence. To solve this problem, we introduce the following state transition equation:

$$\mathbf{N}_{t+1}^{(j)} = (1-\alpha)\mathbf{N}_t^{(j)} + \alpha\hat{\mathbf{N}}_t + \beta\left(\mu_{\mathbf{N}_t}^{(j)} - \mathbf{N}_t^{(j)}\right) + \mathbf{W}_t^{(j)}, \quad (7)$$

where $\hat{\mathbf{N}}_t$, α, and β denote the weighted average of noise sample $\mathbf{N}_t^{(j)}$ calculated by Eq. (8), a forgetting factor, and a scaling factor of feedback, respectively.

$$\hat{\mathbf{N}}_t = \sum_{j=1}^{J} w_t^{(j)} \mathbf{N}_t^{(j)} \ . \quad (8)$$

The first and second terms of Eq. (7) move noise sample $\mathbf{N}_t^{(j)}$ close to the weighted average using the forgetting factor. Thus, this reduces the scatter of samples and can prevent the appearance of outlier samples.

In the third term of Eq. (7), $\mu_{\mathbf{N}_t}^{(j)}$, which is calculated by Eq. (9), is the average of the preceding T samples (Polyak average [10]). The third term of Eq. (7) shows the feedback of the Polyak average and represents the compensation range for parameter prediction to the next time by considering the difference between the average of the preceding samples and the current sample [6].

$$\mu_{\mathbf{N}_t}^{(j)} = \frac{1}{T} \sum_{s=t-T+1}^{t} \mathbf{N}_s^{(j)} \quad (9)$$

Figure 2 illustrates examples of Polyak averaging. When $\mathbf{N}_t^{(j)}$ varies slowly as shown in case (a) of Figure 2, the difference between Polyak average $\mu_{\mathbf{N}_t}^{(j)}$ and $\mathbf{N}_t^{(j)}$ usually has a small value. Thus, the compensation range for parameter prediction from $\mathbf{N}_t^{(j)}$ to $\mathbf{N}_{t+1}^{(j)}$ is regarded as small. On the other hand, when $\mathbf{N}_t^{(j)}$ varies rapidly, as

shown in case (b), the difference between $\mu_{\mathbf{N}_t}^{(j)}$ and $\mathbf{N}_t^{(j)}$ usually has a large value. Thus, the compensation range for parameter prediction from $\mathbf{N}_t^{(j)}$ to $\mathbf{N}_{t+1}^{(j)}$ is regarded as large.

From these facts, it can be assumed that Polyak averaging controls the compensation range for parameter prediction to the next time. A Polyak averaging-based state transition equation can estimate the noise sequence more accurately than the random walk process-based state transition equation because it predicts the next frame parameter depending on varying ranges of preceding frames.

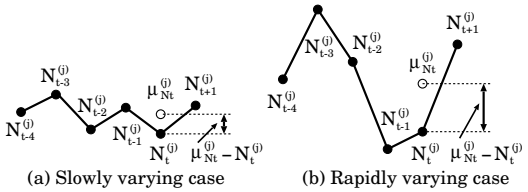

Fig. 2. Examples of Polyak averaging

2.3. Switching dynamical system

The Polyak averaging described in Section 2.2 predicts the $\mathbf{N}_t^{(j)}$ of a future frame depending on the time varying characteristics of the preceding frames. In previous work [9], the parameters of Polyak averaging are fixed as time constants. However, time varying characteristics of non-stationary noise may change frame by frame, therefore, the parameters of Polyak averaging should be set as time variable parameters. For the implementation of this scheme, we introduce a switching dynamical system [11] into the particle filtering.

The switching dynamical system has several dynamical systems with different parameter settings, and switches suitable parameters for the next frame according to the current state $m_t^{(j)}$. The target state is randomly selected according to the state transition probability from current state $m_t^{(j)}$ to target state $n_{t+1}^{(j)}$. In this study, we defined the state transition probability $a_{mn,t}^{(j)}$ as follows.

Figure 3 shows an example of the state transition of the forgetting factor α. In the figure, α has four states (1 to 4), and the value of α (0.05 to 0.20) is output from the corresponding state. When the current state $m_t^{(j)}$ is assigned, the distance between current state $m_t^{(j)}$ to target state $n_{t+1}^{(j)}$ is calculated as the absolute difference of the state index, i.e., $d_{mn,t}^{(j)} = \left| n_{t+1}^{(j)} - m_t^{(j)} \right|$.

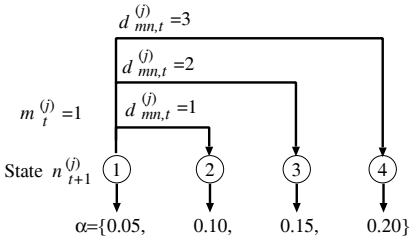

Fig. 3. An example of the state transition of parameter α

The state transition probability is calculated by Eq. (10) using $d_{mn,t}^{(j)}$ and sensitivity parameter γ ($1 \geq \gamma > 0$). After the computation of $a_{mn,t}^{(j)}$, it is normalized as $\sum_n a_{mn,t}^{(j)} = 1$.

$$a_{mn,t}^{(j)} = \gamma^{d_{mn,t}^{(j)}} \tag{10}$$

Eq. (10) means that the state transition probability becomes small when the difference between the current state index and the target state index becomes large. When the target state is the same as the current state, namely the self-loop case, the state transition probability is maximized. The parameter γ controls the sensitivity of state transition probabilities. When γ is set as a small value, the sensitivity of the state transition probabilities becomes sharp. In the case of $\gamma = 1$, equal probabilities are assigned to all of the states.

The calculation of $a_{mn,t}^{(j)}$ and state selection is independently applied to each parameter of Polyak averaging, i.e., α, β, and T.

3. EXPERIMENTS

3.1. Experimental setup

Speaker independent Japanese digit recognition was carried out using HTK ver. 3.2 [14]. The training and testing data were 8,440 connected digit utterances spoken by 110 speakers (55 males and 55 females) and 1,001 connected digit utterances spoken by 104 speakers (52 males and 52 females). These materials were taken from AURORA-2J [12]. Factory and road cutting noises recorded in real environments [15] were artificially added to clean testing data with SNRs from 0 to 20 dB.

The feature parameters used in this evaluation were composed of 39 MFCCs with 13 MFCCs (with zero-th MFCC) and their first and second order derivatives. A zero-th MFCC was used as the energy coefficient instead of a standard Log-energy. At the feature extraction stage, CMS was applied to each sentence.

AURORA-2J standard whole word HMMs (16 states, 20 mixture distributions per state) were used for speech recognition and trained using clean training data. We also trained the clean speech GMM with 512 mixture distributions for MMSE-based noise suppression and particle filter-based sequential noise estimation by using the clean training data of AURORA-2J. The feature parameters were the 23rd order log output energy of a Mel-filter bank.

In particle filter-based noise estimation, the number of samples was fixed at 50, and the covariance matrix of driving noise $\mathbf{W}_t$ was set to $\mathbf{\Sigma}_\mathbf{W} = diag(0.0001)$. The parameters of Polyak averaging and feedback have four states, i.e., $\alpha = \{0.05, 0.10, 0.15, 0.20\}$, $\beta = \{0.5, 1.0, 1.5, 2.0\}$, and $T = \{5, 10, 15, 20\}$, respectively.

3.2. Experimental results

Tables 1 and 2 indicate the speech recognition results for word accuracy. In the tables, "HTK," "ETSI," "MMSE," 'SEM," "PF 1," "PF 2," and "PF 3" indicate the baseline results without noise compensation, results by ETSI Advanced front-end, results by MMSE estimation with stationary noise compensation, results by the sequential EM-algorithm, results by particle filtering with the random walk process, results by particle filtering with Polyak averaging and feedback (constant parameter), and results by particle filtering with switching dynamical system, respectively. In "PF 2," the parameters were set as: factory noise case $\alpha = 0.20$, $\beta = 0.5$, and $T = 10$, and road cutting noise case $\alpha = 0.20$, $\beta = 0.5$, and $T = 20$. In "PF 3," the

Table 1. Word accuracy of factory noise environments (%)

SNR	HTK	ETSI	MMSE	SEM	PF 1	PF 2	PF 3
20 dB	93.61	92.88	96.41	96.50	96.13	96.71	**97.54**
15 dB	81.12	86.86	88.92	90.55	90.02	91.74	**93.43**
10 dB	54.81	76.73	74.27	75.07	75.87	82.13	**85.08**
5 dB	29.47	53.18	50.94	55.60	54.50	64.02	**68.41**
0 dB	18.73	23.15	24.72	26.48	28.92	38.66	**43.20**
Ave.	55.55	66.56	67.05	68.88	69.09	74.65	**77.48**

Table 2. Word accuracy of road cutting noise environments (%)

SNR	HTK	ETSI	MMSE	SEM	PF 1	PF 2	PF 3
20 dB	96.68	96.90	99.20	99.10	98.34	99.26	**99.72**
15 dB	89.93	94.81	97.61	97.90	95.61	98.28	**98.83**
10 dB	70.28	89.81	91.77	92.43	89.84	94.66	**96.93**
5 dB	38.81	76.02	71.57	77.20	75.28	81.79	**86.15**
0 dB	22.29	48.48	43.60	51.00	49.43	58.00	**63.89**
Ave.	63.60	81.20	80.75	83.53	81.70	86.40	**89.10**

Table 3. Average word accuracy with various parameter settings of the switching dynamical system for factory noise environments (%)

SNR	$\gamma = 0.1$	$\gamma = 0.2$	$\gamma = 0.3$	$\gamma = 0.4$	$\gamma = 0.5$	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$	$\gamma = 0.9$	$\gamma = 1.0$
20 dB	97.39	97.45	97.42	97.45	97.54	97.39	97.30	97.36	97.61	**97.82**
15 dB	92.91	93.09	92.88	92.78	93.43	93.15	**93.55**	93.49	**93.55**	93.52
10 dB	84.43	84.31	84.43	**85.08**	84.80	84.99	84.77	84.56	84.71	85.05
5 dB	67.95	67.76	68.07	67.70	**68.41**	68.31	68.10	67.67	68.25	67.82
0 dB	42.31	42.16	42.55	42.31	**43.20**	42.80	42.43	42.74	43.14	42.43
Ave.	77.00	76.95	77.07	77.06	**77.48**	77.33	77.23	77.16	77.45	77.33

Table 4. Average word accuracy with various parameter settings of the switching dynamical system for road cutting noise environments (%)

SNR	$\gamma = 0.1$	$\gamma = 0.2$	$\gamma = 0.3$	$\gamma = 0.4$	$\gamma = 0.5$	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$	$\gamma = 0.9$	$\gamma = 1.0$
20 dB	99.72	99.72	99.66	99.72	99.66	99.72	99.63	99.69	**99.75**	99.72
15 dB	98.99	98.89	98.99	98.93	99.08	98.83	**99.14**	98.86	98.96	99.02
10 dB	97.05	96.96	**97.18**	97.18	97.11	96.93	96.96	97.02	**97.18**	96.87
5 dB	86.18	86.03	86.12	85.75	85.97	86.15	**86.37**	85.54	85.94	85.88
0 dB	63.37	63.83	63.34	63.43	63.43	**63.89**	62.97	63.40	63.59	63.37
Ave.	89.06	89.09	89.06	89.00	89.05	**89.10**	89.01	88.90	89.08	88.97

parameters were set as: factory noise case $\gamma = 0.5$, and road cutting noise case $\gamma = 0.6$.

In the tables, we can see that the results of "PF 3" exhibit the best average word accuracy. They especially showed significant improvement from "PF 2." These results suggest the effectiveness and importance of the parameter switching of Polyak averaging depending on the state transition characteristics of non-stationary noise.

The processing performance of "PF 3" was approximately 1.0 times that of real-time by a 3.2 GHz Intel Xeon processor. On the other hand, The processing performance of "SEM" were approximately 2.0 to 4.0 times that of real-time by the same CPU[1]. From this fact, we can confirm that the proposed method, "PF 3", achieves good speech recognition accuracy and fast processing compared with the conventionally used "SEM".

Tables 3 and 4 indicate word accuracies with various parameter settings of the switching dynamical system ("PF 3"). In all of the conditions, the value of γ that shows the best word accuracy depends on the noise type and SNR type. However, the differences of word accuracies according to γ are less than 1%. Thus, these results suggest that the sensitivity of γ is not a serious problem.

This study assumes that the state transition of the parameter simply depends on the state of the previous frame. However, to represent the accurate state transition, it is necessary to consider the effect of further preceding frames. This may be one of the most important things for sequential noise parameter estimation.

4. CONCLUSION

A particle filter-based non-stationary noise estimation method was presented in this paper. In the proposed method, we introduce a switching dynamical system into particle filtering to cope with the state transition characteristics of non-stationary noise. In the evaluation results, we observed that the switching dynamical system significantly improved speech recognition accuracy in non-stationary noise compared with a dynamical system with time constant Polyak averaging and feedback. In the future, we are planning to investigate the switching dynamical system with consideration of the effect of the further preceding frames, and applications to the sequential estimation of acoustic transfer characteristics, such as the characteristics of moving speech sources.

Acknowledgements

This research was supported in part by the National Institute of Information and Communications Technology. The present study was conducted using a AURORA-2J database developed by the IPSJ-SIG SLP Noisy Speech Recognition Evaluation Working Group.

[1] In "SEM", the number of iterations required for conversion depends on the data. Thus, the processing performance changes depending on the data.

5. REFERENCES

[1] S. F. Boll, "Suppression of Acoustic Noise in Speech Using Spectral Subtraction," *IEEE Trans. on ASSP*, Vol. 27, No. 2, pp. 113-120, Apr. 1979.

[2] ETSI ES 202 050 V1.1.3, "Speech Processing, Transmission and Quality Aspects (STQ); Distributed Speech Recognition; Advanced Front-end Feature Extraction Algorithm; Compression Algorithms," Nov. 2003.

[3] J. C. Segura, A. de la Torre, M. C. Benitez, and A. M. Peinado, "Model-Based Compensation of the Additive Noise for Continuous Speech Recognition. Experiments Using AURORA II Database and Tasks," *Proc. EuroSpeech '01*, Vol. I, pp. 221-224, Sept. 2001.

[4] M. Afify and O. Siohan, "Sequential Estimation with Optimal Forgetting for Robust Speech Recognition," *IEEE Trans. on SAP*, Vol. 12, No. 1, pp. 19-26, Jan. 2004.

[5] K. Yao, K. K. Paliwal, and S. Nakamura, "Noise Adaptive Speech Recognition Based on Sequential Noise Parameter Estimation," *Speech Communication*, Vol. 42, Issue 1, pp. 5-23, Jan. 2004.

[6] T. A. Myrvoll and S. Nakamura, "Online Cepstral Filtering Using a Sequential EM Approach with Polyak Averaging and Feedback," *Proc. ICASSP '05*, Vol. I, pp. 261-264, March, 2005.

[7] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Trans. SP*, Vol. 50, No. 2, pp. 174.188, Feb. 2002.

[8] K. Yao and S. Nakamura, "Sequential noise compensation by sequential Monte Carlo method," *Proc. NIPS '01*, pp. 1205-1212, Dec. 2001.

[9] M. Fujimoto and S. Nakamura, "Particle Filtering and Polyak Averaging-based Non-stationary Noise Tracking for ASR in Noise," *Proc. ASRU '05*, pp. 337-342, Nov. 2005.

[10] H. J. Kushner and J, Yang, "Stochastic Approximation with Averaging and Feedback: Rapidly Convergent "On-Line" Algorithm," *IEEE Trans. on AC*, Vol. 40, No. 1, pp. 24-34, Jan. 1995.

[11] J. Droppo and A. Acere, "Noise Robust Speech Recognition with a Switching Linear Dynamic Model," *Proc. ICASSP '04*, Vol. 1, pp. 953-956, May. 2004.

[12] S. Nakamura, K. Takeda, K. Yamamoto, T. Yamada, S. Kuroiwa, N. Kitaoka, T. Nishiura, A. Sasou, M. Mizumachi, C. Miyajima, M. Fujimoto, and T. Endo, "AURORA-2J, An Evaluation Framework for Japanese Noisy Speech Recognition," *IEICE Transactions on Information and Systems*, Vol. E88-D, No. 3, pp. 535-544, Mar. 2005.

[13] W. K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, Vol. 57, No. 1, pp. 97-109, Jan. 1970.

[14] HTK Web site, http://htk.eng.cam.ac.uk/

[15] T. Endo, T. Horiuchi, T. Shimizu, and S. Nakamura, "Speech Recognition Experiments with ATR Ambient Noise Sound Database – ATRANS –," *IPSJ SIG Technical Reports*, SLP-57-8, pp. 43-48, July 2005. (in Japanese)