PARTICLE FILTER BASED NON-STATIONARY NOISE TRACKING FOR ROBUST SPEECH RECOGNITION

Masakiyo Fujimoto and Satoshi Nakamura

ATR Spoken Language Translation 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 the main 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 front-end processing of speech recognition in noise. In the proposed method, a noise sequence is estimated through a sequential importance sampling step, then a residual resampling step, and finally a Markov chain Monte Carlo step with Metropolis-Hastings sampling. The estimated noise sequence is applied to MMSE-based clean speech estimation method. The evaluations were 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 non-stationary noise environments over noise compensation with stationary noise assumptions.

1. INTRODUCTION

Noise robustness is one of the most important problems for the applications of speech recognition techniques in real environments. For this problem, when adverse noise is restricted to stationary noise, a lot of research in speech recognition in noise has been reported [1]-[4]. However, most of the noise observed in real environments have non-stationary characteristics. To improve the 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 when speech recognition is performed is the only noise added to the speech signal. So both clean speech and noise have non-stationary characteristics.

To solve such problems, several estimation methods of non-stationary noise sequences based on a sequential EM algorithm are reported [5]-[7], that can estimate noise sequence effectively. However, their computation costs are expensive because frame by frame iterative estimation is required to converge noise parameters. Recently, a particle filter-based sequential estimation method [8, 9] 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 required.

In this paper, we present a sequential non-stationary noise estimation method based on a particle filter. In the proposed method, the sequence of non-stationary noise is estimated through extended Kalman filter-based sequential importance sampling, residual resampling, and a Markov chain Monte Carlo with Metropolis-Hastings sampling. Then, noise estimation is carried out frame by frame, and the estimated noise sequence is applied to a minimum mean square error (MMSE)-based clean speech estimation method [2].

A particle filter-based method similar to our proposed method was also reported by Yao et al [9]. Yao's approach is an HMM composition-based acoustic model compensation method which updates an acoustic model sequentially by using estimated noise. On the other hand, our approach is a noise compensation method for front-end processing of speech recognition. Therefore, our method can carry out such multiple stage processing as a combination of front-end noise suppression and acoustic model adaptation for residual noise caused by front-end processing. Moreover, in a HMM composition method, a noise-compensated HMM is composed by using a clean speech HMM and a noise HMM. Generally, both clean speech and noise HMMs used for a HMM composition method are trained by using data without cepstral mean subtraction (CMS). Therefore, a HMM composition method cannot be applied to the CMS for the testing (observation) data.

On the other hand, the proposed method can be applied to CMS to estimated clean speech because it is a noise compensation method for front-end processing, and the acoustic model is trained by using estimated clean speech after CMS processing.

Our proposed method was evaluated on a connected Japanese digits recognition task [10], 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 non-stationary noise environments over noise compensation with stationary noise assumptions.

2. PARTICLE FILTER-BASED NOISE ESTIMATION

2.1. Dynamical system for noise sequence

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.

Let $\mathbf{X}_t$, $\mathbf{S}_t$, and $\mathbf{N}_t$ denote the vectors at t-th short time frame which have logarithmic output energy of Mel-filter bank of observed noisy speech, clean speech, and noise, respectively. The dynamical system for noise sequence is represented as follows.

First, assume that clean speech $\mathbf{S}_t$ can be modeled by a hidden Markov model (HMM). At time t parameter $\mathbf{S}_{s_t,k_t,t}$ is generated from a Gaussian distribution k_t contained in state s_t of clean speech HMM. In this case, the observation process of $\mathbf{X}_t$ can be modeled by the following equation by using $\mathbf{N}_t$ and error signal $\mathbf{V}_t$,

$$\begin{aligned}\mathbf{X}_t &= \mathbf{S}_{s_t,k_t,t} + \log\left(\mathbf{I} + \exp\left(\mathbf{N}_t - \mathbf{S}_{s_t,k_t,t}\right)\right) + \mathbf{V}_t \\ &= f\left(\mathbf{S}_{s_t,k_t,t}, \mathbf{N}_t\right) + \mathbf{V}_t \end{aligned} \quad (1)$$

$$\mathbf{V}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_{\mathbf{S},s_t,k_t}), \quad (2)$$

where $\mathbf{\Sigma}_{\mathbf{S},s_t,k_t}$ denotes the diagonal covariance matrix of Gaussian distribution contained in clean speech HMM.

On the other hand, we assumed that the state transition process of $\mathbf{N}_t$ can be modeled by a random walk process as follows:

$$\mathbf{N}_t = \mathbf{N}_{t-1} + \mathbf{W}_t \quad (3)$$

$$\mathbf{W}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_{\mathbf{W}}), \quad (4)$$

where $\mathbf{W}_t$ and $\mathbf{\Sigma}_{\mathbf{W}}$ denote the driving noise for state transition process and diagonal covariance matrix of $\mathbf{W}_t$, respectively.

2.2. Sequential importance sampling for particle filtering

When a dynamical system derived by Eqs. (1) and (3) is given, *a posteriori* probability density function (PDF) for the sequence of $\mathbf{N}_t$ can be represented by a first order Markov chain as follows:

$$p(\mathbf{N}_{0:t}|\mathbf{X}_{0:t}) = p(\mathbf{N}_0|\mathbf{X}_0) \prod_{t'=1}^{t} p(\mathbf{N}_{t'}|\mathbf{N}_{t'-1}) p(\mathbf{X}_{t'}|\mathbf{N}_{t'}), \quad (5)$$

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 which recursively maximizes the above PDF. In a particle filtering algorithm, *a posteriori* PDF $p(\mathbf{N}_{0:t}|\mathbf{X}_{0:t})$ is approximated by Monte Carlo sampling as follows:

$$p(\mathbf{N}_{0:t}|\mathbf{X}_{0:t}) \simeq \sum_{j=1}^{J} w_t^{(j)} \delta\left(\mathbf{N}_{0:t} - \mathbf{N}_{0:t}^{(j)}\right), \quad (6)$$

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

When direct sampling from $p(\mathbf{N}_{0:t}|\mathbf{X}_{0:t})$ is intractable, we can draw the samples from the other distribution $q(\mathbf{N}_{0:t}|\mathbf{X}_{0:t})$, which is related to $p(\mathbf{N}_{0:t}|\mathbf{X}_{0:t})$ and called an *importance density*. Therefore, when samples $\mathbf{N}_{0:t}^{(j)}$ were drawn from the importance density $q(\mathbf{N}_{0:t}^{(j)}|\mathbf{X}_{0:t})$, sample weight $w_t^{(j)}$ is defined as

$$w_t^{(j)} \propto \frac{p(\mathbf{N}_{0:t}^{(j)}|\mathbf{X}_{0:t})}{q(\mathbf{N}_{0:t}^{(j)}|\mathbf{X}_{0:t})}, \quad (7)$$

where $q(\mathbf{N}_{0:t}^{(j)}|\mathbf{X}_{0:t}) \propto p(\mathbf{N}_{0:t}^{(j)}|\mathbf{X}_{0:t})$.

By applying a Bayesian rule to Eq. (5), *a posteriori* PDF with a recursive formula for obtaining $p(\mathbf{N}_{0:t}|\mathbf{X}_{0:t})$ from $p(\mathbf{N}_{0:t-1}|\mathbf{X}_{0:t-1})$ is given by

$$\begin{aligned} p(\mathbf{N}_{0:t}|\mathbf{X}_{0:t}) &= \frac{p(\mathbf{N}_t|\mathbf{N}_{t-1}) p(\mathbf{X}_t|\mathbf{N}_t)}{p(\mathbf{X}_t|\mathbf{X}_{0:t-1})} p(\mathbf{N}_{0:t-1}|\mathbf{X}_{0:t-1}) \\ &\propto p(\mathbf{N}_t|\mathbf{N}_{t-1}) p(\mathbf{X}_t|\mathbf{N}_t) p(\mathbf{N}_{0:t-1}|\mathbf{X}_{0:t-1}). \end{aligned} \quad (8)$$

Furthermore, if importance density $q(\mathbf{N}_{0:t}|\mathbf{X}_{0:t})$ has the following recursive formula,

$$q(\mathbf{N}_{0:t}|\mathbf{X}_{0:t}) = q(\mathbf{N}_t|\mathbf{N}_{0:t-1}, \mathbf{X}_{0:t}) q(\mathbf{N}_{0:t-1}|\mathbf{X}_{0:t-1}), \quad (9)$$

sample weight $w_t^{(j)}$ can be represented as a following recursive formula by substituting Eq. (9) and Eq. (8) into Eq. (7):

$$w_t^{(j)} \propto w_{t-1}^{(j)} \frac{p(\mathbf{N}_t^{(j)}|\mathbf{N}_{t-1}^{(j)}) p(\mathbf{X}_t|\mathbf{N}_t^{(j)})}{q(\mathbf{N}_t^{(j)}|\mathbf{N}_{0:t-1}^{(j)}, \mathbf{X}_{0:t})}. \quad (10)$$

In Eq. (10), we assumed that $p(\mathbf{N}_t^{(j)}|\mathbf{N}_{t-1}^{(j)}) = q(\mathbf{N}_t^{(j)}|\mathbf{N}_{0:t-1}^{(j)}, \mathbf{X}_{0:t})$. Therefore, it is simplified as

$$w_t^{(j)} \propto w_{t-1}^{(j)} p(\mathbf{X}_t|\mathbf{N}_t^{(j)}), \quad (11)$$

where

$$p(\mathbf{X}_t|\mathbf{N}_t^{(j)}) = \mathcal{N}\left(\mathbf{X}_t; f\left(\mathbf{S}_{s_t^{(j)},k_t^{(j)},t}^{(j)}, \mathbf{N}_t^{(j)}\right), \mathbf{\Sigma}_{\mathbf{S},s_t^{(j)},k_t^{(j)}}\right). \quad (12)$$

Generally, the above particle filtering algorithm is called a sequential importance sampling (SIS) particle filter [8].

2.3. Parameter updating by extended Kalman filter

To update the noise samples $\mathbf{N}_t^{(j)}$ from previous samples $\mathbf{N}_{t-1}^{(j)}$, an extended Kalman filter, which is derived by the dynamical system defined by Eqs. (1) and (3), is applied. The extended Kalman filter-based updating formula is given by

$$\mathbf{N}_{t|t-1}^{(j)} = \hat{\mathbf{N}}_{t-1}^{(j)} \quad (13)$$

$$\mathbf{\Sigma}_{\mathbf{N}t|t-1}^{(j)} = \mathbf{\Sigma}_{\mathbf{N}t-1}^{(j)} + \mathbf{\Sigma}_{\mathbf{W}} \quad (14)$$

$$\mathbf{K}_t^{(j)} = \mathbf{\Sigma}_{\mathbf{N}t|t-1}^{(j)} \mathbf{F}_t^{(j)T} \left[\mathbf{F}_t^{(j)} \mathbf{\Sigma}_{\mathbf{N}t|t-1}^{(j)} \mathbf{F}_t^{(j)T} + \mathbf{\Sigma}_{\mathbf{S},s_t^{(j)},k_t^{(j)}}\right]^{-1} \quad (15)$$

$$\mathbf{F}_t^{(j)} = \partial f\left(\mathbf{S}_{s_t^{(j)},k_t^{(j)},t}^{(j)}, \mathbf{N}_{t|t-1}^{(j)}\right) \Big/ \partial \mathbf{N}_{t|t-1}^{(j)} \quad (16)$$

$$\hat{\mathbf{N}}_t^{(j)} = \mathbf{N}_{t|t-1}^{(j)} + \mathbf{K}_t^{(j)} \left(\mathbf{X}_t - f\left(\mathbf{S}_{s_t^{(j)},k_t^{(j)},t}^{(j)}, \mathbf{N}_{t|t-1}^{(j)}\right)\right) \quad (17)$$

$$\mathbf{\Sigma}_{\mathbf{N}t-1}^{(j)} = \mathbf{\Sigma}_{\mathbf{N}t|t-1}^{(j)} - \mathbf{K}_t^{(j)} \mathbf{F}_t^{(j)} \mathbf{\Sigma}_{\mathbf{N}t|t-1}^{(j)}, \quad (18)$$

where subscript $t|t-1$ denotes the predicted parameter from $t-1$-th frame and $\mathbf{S}_{s_t^{(j)},k_t^{(j)},t}^{(j)}$ denotes the clean speech samples drawn from clean speech HMM as follows:

$$\mathbf{S}_{s_t^{(j)},k_t^{(j)},t}^{(j)} \sim \mathcal{N}\left(\mu_{\mathbf{S},s_t^{(j)},k_t^{(j)}}, \mathbf{\Sigma}_{\mathbf{S},s_t^{(j)},k_t^{(j)}}\right) \quad (19)$$

$$s_t^{(j)} \sim a_{\mathbf{S},s_{t-1}^{(j)}s_t}, \quad k_t^{(j)} \sim P_{\mathbf{S},s_t^{(j)},k_t} \quad (20)$$

where $\mu_{\mathbf{S},s_t^{(j)},k_t^{(j)}}$, $a_{\mathbf{S},s_{t-1}^{(j)}s_t}$, and $P_{\mathbf{S},s_t^{(j)},k_t}$ denote the mean vector of $k_t^{(j)}$-th Gaussian distribution contained in state $s_t^{(j)}$ of clean speech HMM, state transition probability from $s_{t-1}^{(j)}$ to s_t, and mixture weight, respectively. Initial noise samples are drawn as

$$\mathbf{N}_0^{(j)} \sim \mathcal{N}\left(\mu_{\mathbf{N}}, \mathbf{\Sigma}_{\mathbf{N}}\right), \quad \mathbf{\Sigma}_{\mathbf{N}0}^{(j)} = \mathbf{\Sigma}_{\mathbf{N}}, \quad (21)$$

where $\mu_{\mathbf{N}}$ and $\mathbf{\Sigma}_{\mathbf{N}}$ denote mean vector and diagonal covariance matrix of initial noise distribution, respectively. $\mu_{\mathbf{N}}$ and $\mathbf{\Sigma}_{\mathbf{N}}$ are estimated by using the first 10 frames of observations.

2.4. Residual resampling (selection) step

In practice, after the sampling step described in Section 2.2, 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 [8] is adopted after the sampling step. Figure 1 illustrates the summary of residual resampling step.

The method avoids degeneracy by discarding those samples with insignificant weights, and to keep the number of samples constant, samples with significant weights are duplicated. Accordingly, the weights after the selection step are also proportionally redistributed.

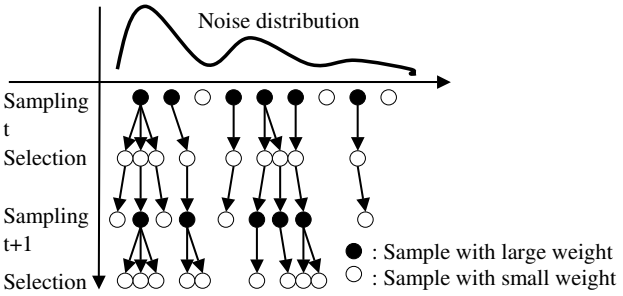

Fig. 1. Summary of residual resampling step

2.5. Markov chain Monte Carlo step

After the resampling step at frame t, these J samples are distributed approximately according to Eq. (6). However, the discrete nature of the approximation can skew the importance weights distribution, where in extreme cases all the samples have the same value.

A Metropolis-Hastings (MH) sampling [11] step is introduced in each sample where the step involves sampling a candidate given the current state according to proposal importance distribution. To simplify the calculation, we assume that the importance distribution is symmetric. After some mathematical manipulation, it is shown that an acceptance possibility is given by

$$\nu = \min\left\{1, w_t^{*(j)}/w_t^{(j)}\right\}, \tag{22}$$

where $w_t^{*(j)}$ denotes sample weight computed by MH sampling step. The state transition by MH sampling is derived as:

$$\mathbf{N}_t^{(j)} = \begin{cases} \mathbf{N}_t^{*(j)} & if\ u \le \nu\ \ (accept\ state\ transition) \\ \mathbf{N}_t^{(j)} & otherwise\ \ (reject\ state\ transition) \end{cases}, \tag{23}$$

where $\mathbf{N}_t^{*(j)}$ denotes samples drawn by MH sampling step and $u \sim U_{[0,1]}$. $U_{[0,1]}$ is the uniform distribution between 0 and 1.

2.6. MMSE estimation of clean speech

We estimated clean speech $\mathbf{S}_t$ by using MMSE estimation [2]. Clean speech estimates with one noise sample are given by the following MMSE formula:

$$\hat{\mathbf{S}}_t^{(j)} = \mathbf{X}_t - \sum_{k=1}^{K} P(k|\mathbf{X}_t,(j)) \log\left(\mathbf{I} + \exp\left(\mathbf{N}_t^{(j)} - \mu_{\mathbf{S},s_t^{(j)},k}\right)\right), \tag{24}$$

where K denotes the number of mixtures and $P(k|\mathbf{X}_t,(j))$ is given by

$$P(k|\mathbf{X}_t,(j)) = \frac{P_{\mathbf{S},s_t^{(j)},k}\mathcal{N}\left(\mathbf{X}_t, \mu_{\mathbf{X}_{k,t}^{(j)}}, \mathbf{\Sigma}_{\mathbf{X}_{k,t}^{(j)}}\right)}{\sum_{k'=1}^{K} P_{\mathbf{S},s_t^{(j)},k'}\mathcal{N}\left(\mathbf{X}_t, \mu_{\mathbf{X}_{k',t}^{(j)}}, \mathbf{\Sigma}_{\mathbf{X}_{k',t}^{(j)}}\right)}, \tag{25}$$

where $\mu_{\mathbf{X}_{k,t}^{(j)}}$ and $\mathbf{\Sigma}_{\mathbf{X}_{k,t}^{(j)}}$ denote mean vector and diagonal covariance matrix of $\mathbf{X}_t$ which are compensated by first order VTS-based approach [4] with parameters $\mu_{\mathbf{S},s_t^{(j)},k}$, $\mathbf{\Sigma}_{\mathbf{S},s_t^{(j)},k}$, $\mathbf{N}_t^{(j)}$ and $\mathbf{\Sigma}_{\mathbf{N}t}^{(j)}$.

Finally, a clean speech estimate $\hat{\mathbf{S}}_t$ is given by

$$\hat{\mathbf{S}}_t = \sum_{j=1}^{J} w_t^{(j)} \hat{\mathbf{S}}_t^{(j)}. \tag{26}$$

3. EXPERIMENTS

3.1. Experimental setup

Speaker independent Japanese connected digits recognition were carried out by using HTK ver.3.2. 8440 connected digits utterances spoken by 110 speakers and 1001 connected digits utterances spoken by 104 speakers were used for training and testing. These materials were taken from AURORA-2J [10]. Factory and road working noises were artificially added to clean testing data with various SNRs from 0dB to 20dB.

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 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. We also trained the clean speech HMM for noise compensation by using clean training data of AURORA-2J with several states and mixture distributions: namely, 1 state model (512 mixtures per state), 4 states model (128 mixtures per state), 8 states model (64 mixtures per state), and 16 states model (32 mixtures per state). The number of distributions contained in an HMM were 512 for all HMMs. The feature parameters were 23th order log output energy of Mel-filter bank. In particle filter-based noise estimation, the number of samples was set to 20, and the covariance matrix of driving noise $\mathbf{W}_t$ was set to $\mathbf{\Sigma}_{\mathbf{W}} = diag(0.01)$.

3.2. Experimental results

Figure 2 illustrates the estimation results of factory noise. In the figure, "True noise," "Moving average," and "Particle filter" indicate the true noise sequence of the first log output energy of a Mel-filter bank, the sequence of "True noise" smoothed by moving average with 20 frame intervals, and, noise sequence estimated by our proposed method with a 16 state HMM, respectively. The processing performance of the proposed method with a 16 state HMM was about 0.8 times that of real time by a 3.2GHz Intel Xeon processor.

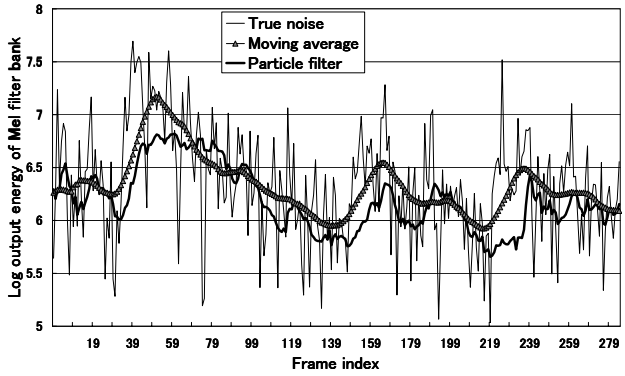

Fig. 2. Estimation results of factory noise by 16 states HMM

Figure 2 shows that the estimated noise sequence is close to that of "Moving average". However, the estimation accuracy was insufficient when abrupt changes occurred. Moreover, compared with "True noise", estimation error was large throughout all noise sequences. This is caused by inaccurate modeling of dynamical

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

SNR	HTK baseline	ETSI Advanced front-end	MMSE (Stationary noise compensation)	Particle filter				Upper limit (True noise)
				1 state	4 states	8 states	16 states	
20 dB	93.61	92.88	**96.41**	96.04	95.03	95.82	96.13	98.46
15 dB	81.12	86.86	88.92	89.59	87.38	88.82	**90.02**	94.87
10 dB	54.81	**76.73**	74.27	75.41	72.95	73.90	75.87	85.11
5 dB	29.47	53.18	50.94	51.98	51.30	50.54	**54.50**	63.13
0 dB	18.73	23.15	24.72	26.19	28.55	26.50	**28.92**	37.06
Average	55.55	66.56	67.05	67.84	67.04	67.12	**69.09**	75.73

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

SNR	HTK baseline	ETSI Advanced front-end	MMSE (Stationary noise compensation)	Particle filter				Upper limit (True noise)
				1 state	4 states	8 states	16 states	
20 dB	96.68	96.90	**99.20**	98.34	97.85	98.46	98.34	99.29
15 dB	89.93	94.81	97.61	**97.08**	95.21	96.07	95.61	98.16
10 dB	70.28	89.81	91.77	**92.39**	88.79	90.33	89.84	94.87
5 dB	38.81	**76.02**	71.57	73.04	72.24	74.21	75.28	82.69
0 dB	22.29	48.48	43.60	43.44	48.88	46.88	**49.43**	54.51
Average	63.60	81.20	80.75	80.86	80.59	81.19	**81.70**	85.90

system for noise. We employed a random walk process represented by Eq. (3) for a state transition process. However, a random walk process does not ensure the accurate state transition of noise. To reduce estimation error of noise sequences, it is necessary to investigate an accurate modeling method of noise dynamics.

Tables 1 and 2 indicate the speech recognition results for word accuracy. In the tables, "HTK Baseline," "ETSI Advanced front-end," "MMSE," and "Particle filter" indicate the recognition results without noise compensation, results by ETSI Advanced front-end [3], results by MMSE estimation with stationary noise compensation and 1 state model, and results by MMSE estimation with particle filter (non-stationary noise compensation), respectively. "Upper limit" indicates the results by MMSE estimation with true noise sequence and 1 state model.

Tables 1 and 2 show that the proposed method with a 16 state model exhibits the best average word accuracy. However, the performance improvement from "MMSE" was small. To improve word accuracy, it is necessary to estimate noise sequences as accurately as possible, since the potential of MMSE with true noise sequence is large.

As we can see from the tables, to improve the accuracy of noise estimation, we have to increase the number of states of clean speech HMMs. These results suggest that noise estimation accuracy by the proposed method depends not only on noise dynamics but also clean speech dynamics, because accurate clean speech samples are required to update the noise samples and compute the sample weight as shown in Sections 2.2 and 2.3. The reliabilities of clean speech samples depend on the modeling accuracy of clean speech HMMs. If a clean speech HMM has a large number of states, the reliabilities of clean speech samples will increase because an HMM with a large number of states can model the detailed dynamics of clean speech. From these facts, we have to consider the optimal modeling of clean speech HMMs for particle filter-based noise estimation.

4. CONCLUSION

A particle filter-based non-stationary noise estimation method has been presented in this paper. In the evaluation results, the proposed method showed improvements of speech recognition accuracy in non-stationary noise environments. Furthermore, it showed that accurate dynamics modeling for both noise and clean speech is an important factor for particle filter-based noise estimation. In the future, we are planning to investigate the accurate modeling of noise dynamics and the optimal modeling of clean speech HMM for particle filter-based noise estimation.

5. 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.

6. REFERENCES

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

[2] 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," EuroSpeech'01, Vol.I, pp.221-224, 2001.

[3] ETSI ES 202 050 V1.1.1, "Distributed speech recognition; advanced front-end feature extraction algorithm; compression algorithms," 2002.

[4] P.J.Moreno, B.Raj, and R.M.Stern: "A Vector Taylor Series Approach for Environment-Independent Speech Recognition," ICASSP'96, pp.733-736, 1996.

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

[6] 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, 2004.

[7] T.A.Myrvoll and S.Nakamura: "Optimal Filtering of Noisy Cepstral Coefficients for Robust ASR," ASRU2003, pp.381-386, 2003.

[8] 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, 2002.

[9] K.Yao and S.Nakamura: "Sequential noise compensation by sequential Monte Carlo method," NIPS2001, pp.1205-1212, 2001.

[10] S.Nakamura, K.Yamamoto, K.Takeda, S.Kuroiwa, N.Kitaoka, T.Yamada, M.Mizumachi, T.Nishiura, M.Fujimoto, A.Sasou, and T.Endo: "Data Collection and Evaluation of AURORA2-J Japanese Corpus," ASRU2003, pp.619-623, 2003.

[11] W.K.Hastings: "Monte Carlo sampling methods using Markov chains and their applications," Biometrika, Vol.57, pp.97-109, 1970.