Bayesian Inference for Speech Density Estimation by the Dirichlet Process Mixture

Kenko Ota [†, ‡], **Emmanuel Duflos** [†], **Philippe VanHeeghe** [†], **and Masuzo Yanagida** [‡]

†LAGIS (UMR CNRS 8146), Ecole Centrale de Lille

BP48, Cité Scientifique, 59651, Villeneuve d'Ascq, France

‡Doshisha University, Dept. of Engineering

1-3, Tatara-Miyakodani, Kyotanabe, Kyoto, 610-0321, Japan

etf1704@mail4.doshisha.ac.jp emmanuel.duflos@ec-lille.fr

philippe.vanheeghe@ec-lille.fr myanadig@mail.doshisha.ac.jp

Abstract: This paper shows a method for the modeling of speech signal distributions based on Dirichlet Process Mixtures (DPM) and the estimation of noise sequences based on particle filtering. In real situations, the speech recognition rate degrades miserably because of the effect of environmental noises, reflected waves and so on. To improve the speech recognition rate, a technique for the estimation of noise sequences is necessary. In this paper, the distribution of the clean speech is modeled using the DPM instead of the traditional model, which is Gaussian Mixture Model (GMM). Speech signal sequences are generated according to the mean and covariance generated from the DPM. Then, noise signal sequences are estimated with a particle filter. The proposed method can improve the speech recognition rate significantly in the low SNR region.

Keywords: speech recognition, Kalman filter, Dirichlet process mixture, density estimation, particle filter.

Kenko Ota graduated from Doshisha University in 2003, and received M.E. degree in 2005 from Doshisha University, Japan. Mr. Ota is currently a Ph.D student at Doshisha University and Ecole Centrale de Lille, France, under a double degree system between them. His research interests include blind signal processing, adaptive signal processing and speech recognition.He received the research incentive award from Kansai-section Joint Convention of Institutes of Electrical Engineering in 2004. He is a student member of ASJ, IPSJ and IEEE.

Emmanuel Duflos was born in Amiens, France, on June 20, 1968. He received the Engineer degree from the Institut Supérieur d'Electronique du Nord (ISEN), Lille, France, in 1991, the Diploma of Advanced Study in automatic control and signal processing from the University of Paris XI, Orsay, France, in 1992, the Ph.D. degree in Engineering from the Université de Toulon et du Var, Toulon, France, in 1995, and the Habilitation à Diriger des Recherches (HDR) from the University of Lille I, Lille, in 2003, respectively. He is currently a Professor with Ecole Centrale de Lille, Laboratoire d'Automatique, Génie Informatique et Signal, Villeneuve d'Ascq, France, after eight years with ISEN. His current research activity deals with multisensor systems from signal analysis for data fusion to multisensor management in moving multitarget environment. He is coauthor (with P.Vanheeghe) of several papers about guidance law modeling, multisensor management systems with application to radar sensor management, and personnel landmine detection.

Philippe VanHeeghe was born in France on July 20, 1956. He received the M.S. degree in data processing, the Diploma of Advanced Study in data processing, the Ph.D. degree, and the Habilitation à Diriger des Recherches (HDR) from the University of Lille, Lille, France, in 1981, 1982, 1984, and 1996, respectively. He is currently a Professor with Ecole Centrale de Lille, Lille. He was an Assistant Professor with the Institut Supérieur d'Electronique du Nord, Lille, where he became Head of the Signals and Systems Department in 1990. He is Head of the Laboratoire d'Automatique, Génie Informatique et Signal (LAGIS UMR CNRS), Villeneuve d'Ascq, France. His research activities include multisensor management, signal processing, signals, and systems modeling. He is a coauthor (with E. Duflos) of several papers about guidance law modeling, multisensor management systems with application to radar sensor management, and personnel landmine detection. He has been a Session Organizer for many international conferences. Dr. Vanheeghe has been a Member of the International Program Committee of several international symposiums (IEEE, IMACS).

Masuzo Yanagida graduated from Osaka University in 1969, and received M.E. degree in 1971 and Doctoral degree in 1978, respectively, both from Osaka University, Japan. He worked for only a short time with Japan Broadcasting Corporation and returned to Osaka University in 1972 as a research associate at the Institute of Scientific and Industrial Research. He worked with ISIR, Osaka University for 9 years from 1978 to 1987 and moved to Radio Research Laboratory (now, NICT-MIAC after CRL-MPT), Japan, in 1987. He has been working with Doshisha University since 1994. He received the Sato Paper Award from the Acoustical Society of Japan in 2004, and Society Award from Information Systems Society of the Institute of Electronics, Information and Communication Engineers, Japan in 2006. Dr. Yanagida is currently the Chair of Musical Acoustics Group, Acoustical Society of Japan, and is a member of ASJ, IEICE, IPSJ, JSAI, JCSS, SOFT, IEEE and ASA.

1. Introduction

Currently, noise robustness is one of the most important problems for developing the effective speech recognition systems in real environments. Several techniques using array microphone are proposed, e.g. delay-and-sum array [14], Griffith-Jim array [18] etc. in order to improve speech recognition rate in real environments. Moreover, as a different approach, Independent Component Analysis [9] attracted the interest in order to solve the Blind Source Separation problem.

On the other hand, S.F. Boll proposed Spectral Subtraction [6] as a technique with a single microphone. In general, the techniques with a single microphone demand the accurate noise estimation. It is not difficult to accurately estimate the noise sequence of stationary noise, e.g. white noise. However, many non-stationary noises, e.g. TV set sound or human voice etc., exist in real environments. So, it is difficult to improve speech recognition rate using simple Spectral Subtraction.

Owing to the advancement of computer performance, a particle filtering [3] attracts the attention and is applied to various research fields. Within the field of speech recognition, Fujimoto *et al.* proposed a noise tracking technique based on a particle filtering [15]. This technique consists of the following two parts: one is a noise estimation based on particle filtering and the other is a minimum mean square error (MMSE) based estimation with a Gaussian Mixture Model (GMM) of the clean speech. An essential point of this technique is to develop an accurate GMM beforehand. To develop the accurate GMM it is necessary to use huge number of speech data.

This paper proposes a technique for the estimation of noise and speech sequences without developing the GMM. Instead of the GMM, the speech distribution is modeled using a DPM [2]. The Dirichlet Process (DP) [13] is a non-parametric probability distribution over the space of all possible distributions. The DP is used as the prior of the DPM. The DP can be considered as the probability distribution for the probability distribution of mixture components. The DP is a generative model for infinite distribution. So, DPM allows us to mix the infinite probability distribution. By using DPM in the estimation process of the clean speech distribution, it is expected to estimate this distribution more flexibly.

There are several researches on the nonparametric density estimation using DPM [12], [19]. Caron *et al.* [7] applied the DPM to the density estimation in the context of dynamic models. Caron *et al.* can achieve the improvement of the performance of standard algorithms when the noise pdfs are unknown. Hence, in case where the clean speech distributions are unknown, we also expect to get better result than the standard algorithms.

This paper is organized in the following seven sections: the section 2 describes the Bayesian algorithms for linear/nonlinear filtering, the section 3 declares the problem solved, the section 4 describes the traditional method for modeling the clean speech, the section 5 describes the proposed method, the section 6 shows the evaluation of the proposed method on the speech recognition and section 7 concludes this paper.

2. Bayesian Algorithms for Linear/Nonlinear Filtering

The objective of the dynamic state estimation by the Bayesian approach is to construct the *posterior* probability density function (pdf) $p(n_t \mid x_{1:t})$ based on the observed sequence $x_{1:t} = \{x_1, x_2, \cdots, x_t\}$, where x_t stands for the measurement vector at time *t* and n_t stands for the state vector at time *t*. To define the problem of linear/nonlinear filtering, the state evolves according to the following model:

$$n_t = f_{t-1}(n_{t-1}, w_{t-1}) \tag{1}$$

where f_{t-1} is a known, linear/nonlinear function of the state n_{t-1} and of the process noise w_{t-1}. The measurement is related to the state via the measurement model:

$$x_t = g_t(n_t, v_t) \tag{2}$$

where g_t is a known, linear/nonlinear function and v_t is measurement noise. The pdf $p(n_t \mid x_{1:t})$ is obtained recursively via eqs. (1) and (2) from the pdf $p(n_{t-1} \mid x_{1:t-1})$ in the following two stages: prediction and update [3], [8].

We suppose that the pdf $p(n_{t-1} \mid x_{1:t-1})$ is available. Firstly, at the prediction stage, the prediction density $p(n_t \mid x_{1:t-1})$ of the state at time *t* can be obtained via the following Chapman-Kolmogorov equation:

$$p(n_t \mid x_{1:t-1}) = \int p(n_t \mid n_{t-1}) p(n_{t-1} \mid x_{1:t-1}) dn_{t-1}$$

where the pdf $p(n_t \mid n_{t-1})$ is defined by the eq. (1). Secondly, at the update stage, when the measurement x_t is observed, the updated pdf can be obtained from the prediction pdf via the following Bayesian rule:

$$
\begin{aligned}
p(n_t \mid x_{1:t}) &= p(n_t \mid x_t, x_{1:t-1}) \\
&= \frac{p(x_t \mid n_t, x_{1:t-1}) p(n_t \mid x_{1:t-1})}{p(x_t \mid x_{1:t-1})} \\
&= \frac{p(x_t \mid n_t) p(n_t \mid x_{1:t-1})}{p(x_t \mid x_{1:t-1})}
\end{aligned}
\tag{3}
$$

where the normalizing constant

$$p(x_t \mid x_{1:t-1}) = \int p(x_t \mid n_t) p(n_t \mid x_{1:t-1}) dn_t \tag{4}$$

depends on the likelihood function $p(x_t \mid n_t)$ defined by the eq. (2). In general, the pdfs given by eqs. (3) and (4) cannot be determined analytically. In case where the functions f_{t-1} and g_t are linear and the pdf $p(n_t \mid x_{1:t})$ is Gaussian, an optimal algorithm, Kalman filter, can be formulated. In the other cases, we have to use approximations or suboptimal Bayesian algorithms, Extended Kalman filter, Particle filter. Brief descriptions of these algorithms are presented in the following sections.

2.1 Kalman Filter

The Kalman filter [3], [8] is assumed that the *posterior* pdf at every time is Gaussian and the functions f_{t-1} and g_t are linear. That is, eqs. (1) and (2) can be rewritten as:

$$
\begin{aligned}
n_t &= F_{t-1} n_{t-1} + w_{t-1} \\
x_t &= G_t n_t + v_t
\end{aligned}
$$

where F_{t-1} and G_t are the matrices defining the linear functions, w_{t-1} and v_t are mutually independent zero-mean White Gaussian whose covariances are Q_{t-1} and R_t respectively. The Kalman algorithm, derived by eqs. (3) and (4), can be considered as the following recursive relationships:

$$
\begin{aligned}
p(n_{t-1} \mid x_{1:t-1}) &= \mathcal{N}(n_{t-1}; \hat{n}_{t-1|t-1}, P_{t-1|t-1}) \\
p(n_t \mid x_{1:t-1}) &= \mathcal{N}(n_t; \hat{n}_{t|t-1}, P_{t|t-1}) \\
p(n_t \mid x_{1:t}) &= \mathcal{N}(n_t; \hat{n}_{t|t}, P_{t|t})
\end{aligned}
$$

where $\mathcal{N}(n; m, P)$ is a Gaussian density with argument *n*, the state, mean *m* and covariance *P*. The appropriate mean and covariance of the Kalman filter are computed as follows:

$$
\begin{aligned}
\hat{n}_{t|t-1} &= F_{t-1} \hat{n}_{t-1|t-1} \\
P_{t|t-1} &= Q_{t-1} + F_{t-1} P_{t-1|t-1} F_{t-1}^T \\
\hat{n}_{t|t} &= \hat{n}_{t|t-1} + K_t (x_t - G_t \hat{n}_{t|t-1}) \\
P_{t|t} &= P_{t|t-1} - K_t S_t K_t^T
\end{aligned}
$$

where

$$S_t = G_t P_{t|t-1} G_t^T + R_t$$

is the covariance matrix of $x_t - G_t \hat{n}_{t|t-1}$, and

$$K_t = P_{t|t-1} G_t^T S_t^{-1}$$

is the Kalman gain.

2.2 Extended Kalman Filter

In the real situations, the optimal filter (i.e. Kalman filter) is hard to use because of the nonlinearity of the target state. Instead, we have to use approximations or suboptimal Bayesian algorithms. In this section, we introduce the Extended Kalman Filter (EKF) [3], [8].

The EKF can be applied for nonlinear function f_{t-1} and g_t with additive noise. So, eqs. (1) and (2) can be rewritten as follows:

$$n_t = f_{t-1}(n_{t-1}) + w_{t-1} \tag{5}$$

$$x_t = g_t(n_t) + v_t \tag{6}$$

Then, the nonlinear functions in eqs. (5) and (6) are approximated by the first term in their Taylor series expansion. The mean and covariance of the EKF are computed as follows:

$$\hat{n}_{t|t-1} = f_{t-1}(\hat{n}_{t-1|t-1})$$

$$P_{t|t-1} = Q_{t-1} + \hat{F}_{t-1} P_{t-1|t-1} \hat{F}_{t-1}^T$$

$$\hat{n}_{t|t} = \hat{n}_{t|t-1} + K_t(x_t - g_t(\hat{n}_{t|t-1}))$$

$$P_{t|t} = P_{t|t-1} - K_t S_t K_t^T$$

where

$$S_t = \hat{G}_t P_{t|t-1} \hat{G}_t^T + R_t$$

$$K_t = P_{t|t-1} \hat{G}_t^T S_t^{-1}$$

$\hat{F}_{t-1}$ and $\hat{G}_t$ are the local linearization of functions f_{t-1} and g_t respectively.

$$\hat{F}_{t-1} = \left.\frac{\partial f_{t-1}}{\partial n_{t-1}}\right|_{n_{t-1}=\hat{n}_{t-1|t-1}}$$

$$\hat{G}_{t-1} = \left.\frac{\partial g_t}{\partial n_{t-1}}\right|_{n_{t-1}=\hat{n}_{t|t-1}}$$

2.3 Particle Filter

Particle filter [3], [8], [11] is also a suboptimal filter. The particle filter can be applied to nonlinear and nongaussian problems. In this section, a particle filtering based on the sequential importance sampling is introduced. The fundamental idea of the particle filter is that the posterior density $p(n_{0:t} | x_{1:t})$ are approximated by the particles generated from the importance density.

$$p(n_{0:t} | x_{1:t}) \cong \sum_{j=1}^{J} \omega_t^{(j)} \delta(n_{0:t} - n_{0:t}^{(j)})$$

where j is the particle index, J is the total number of the particles, $\omega_t^{(j)}$ is the particle weight, the particles consist of $\omega_t^{(j)}$ and $n_{0:t}^{(j)}$ and $\delta(\cdot)$ is a delta function. If the samples $n_{0:t}^{(j)}$ are drawn from an importance density $q(n_{0:t} | x_{1:t})$, then the weigth $\omega_t^{(j)}$ is represented as follows:

$$\omega_t^{(j)} \propto \frac{p(n_{0:t}^{(j)} | x_{1:t}^{(j)})}{q(n_{0:t}^{(j)} | x_{1:t}^{(j)})} \tag{7}$$

$\propto$ represents that the left term is proportional to the right term. $p(n_{0:t} | x_{1:t})$ is written by the following recursive formula using the Bayesian rule.

$$
\begin{aligned}
& p(n_{0:t} \mid x_{1:t}) \\
& = \frac{p(x_t \mid n_{0:t}, x_{1:t-1}) p(n_{0:t} \mid x_{1:t-1})}{p(x_t \mid x_{1:t-1})} \\
& = \frac{p(x_t \mid n_{0:t}, x_{1:t-1}) p(n_t \mid n_{0:t-1}, x_{1:t-1}) p(n_{0:t-1} \mid x_{1:t-1})}{p(x_t \mid x_{1:t-1})} \\
& = \frac{p(x_t \mid n_t) p(n_t \mid n_{t-1})}{p(x_t \mid x_{1:t-1})} p(n_{0:t-1} \mid x_{1:t-1}) \\
& \propto p(x_t \mid n_t) p(n_t \mid n_{t-1}) p(n_{0:t-1} \mid x_{1:t-1}) \qquad (8)
\end{aligned}
$$

If $q(n_{0:t} \mid x_{1:t})$ can be expressed by the following recursive formula

$$q(n_{0:t} \mid x_{1:t}) = q(n_t \mid n_{0:t-1}, x_{1:t}) q(n_{0:t-1} \mid x_{1:t-1}) \qquad (9)$$

then sample weight $\omega_t^{(j)}$ can be represented as the following recursive formula by substituting eqs. (8) and (9) into eq. (7)

$$\omega_t^{(j)} \propto \omega_{t-1}^{(j)} \frac{p(x_t \mid n_t^{(j)}) p(n_t^{(j)} \mid n_{t-1}^{(j)})}{q(n_t^{(j)} \mid n_{0:t-1}^{(j)}, x_{1:t})}$$

3. Problem Statement

Now, we want to realize the speech recognition with a single microphone in noisy and reverberant environments. In this problem, the accuracy of noise sequence estimation is one of the most important things. Therefore, we want to estimate not only a clean speech sequence but also a noise sequence. Fujimoto *et al.* dealt with the noise sequence estimation problem using a particle filtering and a clean speech sequence estimation by a GMM. However, by using DPM in the estimation process of the clean speech, it is expected to estimate the clean speech more flexibly.

4. Conventional Method by Fujimoto et al. [15]

In this section, we ignore the effect of the reflected waves. In the frequency domain, we have the following relationship between speech *S* and noise signal *N*:

$$X = S + N$$

where *X* is a observed signal. Speech recognition is generally performed in the log spectral domain. So, if we define *X*=exp(*x*), *S*=exp(*s*) and *N*=exp(*n*), we can get

$$\exp(x) = \exp(s) + \exp(n)$$

$$\log(\exp(x)) = \log(\exp(s) + \exp(n))$$

$$x = s + \log(1 + \exp(n - s))$$

where, *x*, *s* and *n* denote *X*, *S* and *N* in the log spectral domain respectively. The above model has been proposed by Segura *et al.* in [20]. So, it is necessary to consider the nonlinear relationship between the original speech signal and the noise signal. In this conventional method, a particle filter base noise tracking is used.

4.1 Dynamic Model for a Conventional Method

Conventional method is based on the utilization of GMM proposed by Fujimoto *et al.*. Fujimoto *et al.* employed the observed signal model proposed by Segura *et al.* for each particle as follows [20]:

$$\begin{aligned} x_t &= s_{t,k_t} + \log(I + \exp(n_t - s_{t,k_t})) + v_t \\ &= g(s_{t,k_t}, n_t) + v_t \end{aligned} \tag{10}$$

$$n_t = n_{t-1} + w_{t-1} \tag{11}$$

$$v_t \sim \mathcal{N}(0, \Sigma_{s,k_t}),\ \ w_t \sim \mathcal{N}(0, \Sigma_w)$$

where, t is a frame index. A frame is a time interval for performing a short-term Fourier transform. s_{t,k_t} is modeled by a GMM representing as $S = \sum_k P_{s,k} \mathcal{N}(\mu_{s,k}, \Sigma_{s,k})$ and s_{t,k_t} is generated as follows:

$$k_t \sim P_s$$

where k_t is randomly chosen according to the mixture weight vector P_s for each Gaussian distribution and then

$$s_{t,k_t} \sim \mathcal{N}(\mu_{s,k_t}, \Sigma_{s,k_t})$$

where, μ_{s,k_t} and Σ_{s,k_t} denote the mean vector and diagonal covariance matrix of the k_t -th gaussian mixture component.

4.2 Conventional Algorithm

The noise samples and noise covariance, call the parameters in the following, are estimated by a particle filter. When we use the word “the noise sample”, it is from the application point of view not particle filtering point of view. This particle filter consists of an EKF for parameter updating, a sample weight computation [3], residual resampling and a Markov Chain Monte Carlo with Metropolis-Hastings sampling [17] for random variable drawing. The speech signal s is estimated by MMSE estimation. The initial noise sample is drawn as

$$n_0^{(j)} \sim \mathcal{N}(\mu_n, \Sigma_n)$$

$$\Sigma_{n_0}^{(j)} = \Sigma_n$$

where, μ_n and Σ_n denote the mean vector and diagonal covariance matrix of initial noise distribution respectively. μ_n and Σ_n are estimated by the first 5 frames of the observed signal with no clean speech in the observed signal.

The tracking performances of noise sequences depends on the accuracy of GMM. In order to develop an accurate GMM, it takes very long time and needs huge volume of data. It will be a problem for applying to various applications.

4.3 Polyak Averaging and a Switching Dynamical System

In order to improve the performance of noise tracking, Fujimoto *et al.* employ a Polyak averaging and a switching dynamical system [16]. In real situations, noise signal is not always random, so it is necessary to accurately model the noise sequence. The Polyak averaging is expressed as follows:

$$\begin{aligned} n_t^{(j)} &= (1-\alpha_p) n_{t-1}^{(j)} + \alpha_p \hat{n}_{t-1} \\ &\quad + \alpha_p \beta_p (\mu_{n,t}^{(j)} - n_{t-1}^{(j)}) + w_{t-1}^{(j)} \end{aligned}$$

$$\hat{n}_{t-1} = \sum_{j=1}^{J} \omega_t^{(j)} n_{t-1}^{(j)}$$

$$\mu_{n,t}^{(j)} = \frac{1}{T_p} \sum_{s=t-T_p+1}^{t} n_{s-1}^{(j)}$$

In real situations, the aspect of noise fluctuation is also time variant. So, parameters for the Polyak averaging, α_p, β_p and T_p need to change according to time. To realize this mechanism, a switching dynamical system is introduced leading to a Jump Markov System. This switching dynamical system has several dynamical systems with different parameter settings, and switches suitable parameters for the next frame according to the index of the current model $m_t^{(j)}$. The target model at the next time instance is randomly selected according to the transition probability from the current model $m_t^{(j)}$ to the target model $m_{t+1}^{(j)}$. The transition probability is defined as follows:

$$p^{(j)}_{m_t,m_{t+1}} = \gamma^{\left|m^{(j)}_{t+1}-m^{(j)}_t\right|}$$

where the range of γ is $0 \le \gamma \le 1$. Also, the transition probability $p^{(j)}_{m_t,m_{t+1}}$ is normalized as $\sum_{m_{t+1}} p^{(j)}_{m_t,m_{t+1}} = 1$.

4.4 Parameter Updating by Extended Kalman Filter (EKF)

To update the noise parameters an EKF is applied. This EKF is derived from the eqs. (10) and (11).

$$\begin{aligned} n^{(j)}_{t|t-1} &= (1-\alpha_p)n^{(j)}_{t-1} + \alpha_p \hat{n}_{t-1} \\ &\quad + \alpha_p \beta_p (\mu^{(j)}_{n,t} - n^{(j)}_{t-1}) + w^{(j)}_{t-1} \end{aligned} \tag{12}$$

$$\hat{n}_{t-1} = \sum_{j=1}^{J} \omega^{(j)}_t n^{(j)}_{t-1}$$

$$\mu^{(j)}_{n,t} = \frac{1}{T_p} \sum_{s=t-T_p+1}^{t} n^{(j)}_{s-1} \tag{13}$$

$$\Sigma^{(j)}_{n_{t|t-1}} = F^{(j)}_{t-1} \Sigma^{(j)}_{n_{t-1}} F^{(j)T}_{t-1} + \Sigma_w$$

$$K^{(j)}_t = \Sigma^{(j)}_{n_{t|t-1}} G^{(j)T}_t [G^{(j)}_t \Sigma^{(j)}_{n_{t|t-1}} G^{(j)T}_t + \Sigma_{s,k^{(j)}_t}]^{-1}$$

$$F^{(j)}_{t-1} = \alpha_p(-1 + \omega^{(j)}_{t-1} + \frac{\beta_p}{T_p})$$

$$G^{(j)}_t = \frac{\partial}{\partial n^{(j)}_{t|t-1}} g(s^{(j)}_{t,k^{(j)}_t}, n^{(j)}_{t|t-1})$$

$$\hat{n}^{(j)}_t = n^{(j)}_{t|t-1} + K^{(j)}_t (x_t - g(s^{(j)}_{t,k^{(j)}_t}, n^{(j)}_{t|t-1})) \tag{14}$$

$$\Sigma^{(j)}_{n_t} = \Sigma^{(j)}_{n_{t|t-1}} - K^{(j)}_t G^{(j)}_t \Sigma^{(j)}_{n_{t|t-1}} \tag{15}$$

Equations (12) and (13) are equations for the prediction and a Polyak averaging [16] is employed. $K_t^{(j)}$ is the Kalman gain. $G_t^{(j)}$ is the linearization function. Equations (14) and (15) are equations for the update.

4.5 Sequential Importance Sampling for Particle Filtering [3]

In the particle filtering algorithm, *a posteriori* pdf $p(n_{0:t} \mid x_{0:t})$ is approximated by Monte Carlo sampling as follows:

$$p(n_{0:t} \mid x_{0:t}) \cong \frac{1}{J}\sum_{j=1}^{J} \delta(n_{0:t} - n_{0:t}^{(j)})$$
$$\cong \sum_{j=1}^{J} \omega_t^{(j)} p(n_{0:t}^{(j)} \mid x_{0:t})$$

In the sequential importance sampling, sample weight $\omega_t^{(j)}$ can be represented as the following recursive formula

$$\omega_t^{(j)} \propto \omega_{t-1}^{(j)} \frac{p(n_t^{(j)} \mid n_{t-1}^{(j)}) p(x_t \mid n_t^{(j)})}{q(n_t^{(j)} \mid n_{0:t-1}^{(j)}, x_{0:t})} \tag{16}$$

where $q(\cdot \mid \cdot)$ is an *importance density*.

If it is assumed that the pdf $p(n_t^{(j)} \mid n_{t-1}^{(j)})$ is equal to $q(n_t^{(j)} \mid n_{0:t-1}^{(j)}, x_{0:t})$, then the expression of eq.(16) can be rewritten

$$\omega_t^{(j)} \propto \omega_{t-1}^{(j)} p(x_t \mid n_t^{(j)})$$

where

$$p(x_t \mid n_t^{(j)}) = \mathcal{N}(x_t ; g(s_{t,k_t^{(j)}}^{(j)}, n_t^{(j)}), \Sigma_{s,k_t^{(j)}})$$

That is to say, Fujimoto *et al.* employed bootstrap filter.

4.6 Residual Resampling

After calculating sample weights, some of the samples become insignificant. These samples will degenerate the estimation. So, residual resampling step [3] is introduced after the weight calculation. In the residual resampling step, the samples are generate by a resampling with replacement which is proportional to their weights. This method can avoid degeneracy problem by discarding samples with insignificant weights, and to maintain a constant number of samples.

The residual resampling can reduce the effects of degeneracy. However, it causes the other problem which is the particles having high weights are selected many times. As a result, this leads to a loss of diversity among the particles.

4.7 Markov Chain Monte Carlo Step

After the residual resampling step, there is a possibility that most of particles have a same value. To avoid the loss of diversity among the particles, Fujimoto *et al.* introduced a Metropolis-Hasting (MH) sampling [17] in each sample. To simplify the calculation, Fujimoto *et al.* assume that the importance distribution is symmetric. So, the acceptance probability is given by

$$\nu = \min\left\{1, \frac{\omega_t^{*(j)}}{\omega_t^{(j)}}\right\}$$

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

$$\Phi_t^{(j)} = \begin{cases} \Phi_t^{*(j)} & \text{if } u \le \nu \\ \Phi_t^{(j)} & \text{otherwise} \end{cases}$$

where $\Phi_t^{(j)} = (\omega_t^{(j)}, \hat{n}_t^{(j)}, \Sigma_{n_t}^{(j)})$, $\Phi_t^{*(j)}$ is samples drawn by the MH sampling step, or the outputs from EKF, and *u* is drawn from the uniform distribution $0 \le u \le 1$.

4.8 Comment on the Conventional Algorithm

According to us, it seems important to note that Fujimoto *et al.* doesn't use the particle filter in a conventional way. He uses directly the output of the EKF as a new particle. Whereas, the outputs of the filter are usually used as the parameters of a normal law from which we can draw the new particle. In the Fujimoto algorithm, random is introduced in the last step using the MH sampling.

5. Proposed Method

We propose the modeling of the clean speech using DPM instead of GMM. By introducing DPM, we expect more flexible estimation of clean speech. Because DPM allows us to mix infinite probability distribution. Moreover, DPM can adapt automatically the number of gaussian laws needed. If we want to mix other laws than gaussian, it is also possible.

5.1 Density Estimation of the Clean Speech

The density estimation problem can be formulated as a following hierarchical form:

$$\begin{aligned} &\mathrm{G} \sim P(\mathrm{G}) \\ &\theta_t \sim \mathrm{G} \\ &s_t \sim f(\cdot \mid \theta_t) \end{aligned}$$

where G is a Random Probability Measure (RPM), $P(\cdot)$ is *a priori* distribution, θ_t is called the latent variable, $f(\cdot \mid \theta_t)$ is a mixed probability density function and s_t is a clean speech. In this model, the problem is how to define *a priori* distribution. We employ here the RPM following a Dirichlet Process (DP) prior.

5.2 Dirichlet Processes

Ferguson *et al.* [9] defined two properties for the adequate *a priori* distribution for $P(\cdot)$.

1. The support of the prior distribution should be large.
2. Posterior distribution given a sample of observation from the true probability distribution should be manageable analytically.

In [9], the authers introduced the DP as a probability measure on the space of probability measures, which satisfies the above properties. A probability distribution G is drawn from $DP(\mathrm{G}_0, \alpha)$ where a probability measure G_0 is defined on a measurable space $(\Omega, \mathcal{A})$, α is a positive real number called scale factor. The Dirichlet distribution is the unique distribution over the space of all possible distributions on $\mathcal{A}$ and satisfies the following relation

$$(\mathrm{G}(A_1), \cdots, \mathrm{G}(A_k)) \sim \mathcal{D}(\mathrm{G}_0(A_1), \cdots, \mathrm{G}_0(A_k), \alpha)$$

where $\mathcal{D}$ is a Dirichlet distribution and $A_i \in \mathcal{A}$ [8].

Many probability distributions can be obtained using urn models. The urn model that corresponds to the Dirichlet distribution is the Polya urn model [5]. Polya urn model is defined as follows: Consider a bag with α balls. Initially the number of balls of color *j* is m_j. We draw balls at random from the bag and at each step we replace the ball that we drew by two same color balls. Then, the probability of the obtaining a ball of color *j* at the *i*th step $P(X_i = j)$ is represented as follows:

$$P(X_i = j \mid X_{1:i-1}) = \frac{m_j + \sum_{k=1}^{i} \delta(X_k = j)}{\alpha + i}$$

A method for obtaining the Dirichlet process is to consider the limit of the number of colors in the Polya urn model. Moreover, Blackwell *et al.* [5] showed that the predictive distribution is given by the Polya urn model as follows

$$\theta_{t+1} \mid \theta_t \sim \frac{\alpha}{\alpha+t} \mathrm{G}_0 + \frac{1}{\alpha+t} \sum_{j=1}^{t} \delta(\theta - \theta_j)$$

5.3 Dirichlet Process Mixture

It is now possible to reformulate the density estimation problem using the following hierarchical model known as DPM [7]:

$$\mathrm{G} \sim DP(\mathrm{G}_0, \alpha)$$

$$\theta_t \sim \mathrm{G}$$

$$s_t \sim f(\cdot \mid \theta_t)$$

where the RPM G is the mixture distribution distributed according to $DP(\mathrm{G}_0, \alpha)$. The latent variables θ_t are distributed according to G. $f(\cdot \mid \theta_t)$ is a mixed probability density function. The following flexible model is adopted for the unknown distribution F

$$F(s) = \int_{\Theta} f(s \mid \theta) d\mathrm{G}(\theta)$$

with $\theta \in \Theta$.

5.4 Estimation of Speech Signal Distribution with the Dirichlet Process Mixture

In the bayesian framework, our problem of estimating a noise sequence and a clean speech sequence, is equivalent to the determination of the probability $p(n_{0:t}, s_{1:t} \mid x_{1:t})$. A clean speech s_t is supposed to be distributed according to a DPM of base mixed distribution $\mathcal{N}(\mu_t, \Sigma_t)$ and scale parameter α [7]. Instead of developing an accurate GMM, we introduce the estimation of clean speech signal distribution with the DPM which will adapt automatically the number of Gaussian laws to use for the modeling of the clean speech. The problem is now to determine the probability $p(n_{0:t}, \theta_{1:t} \mid x_{1:t})$, decomposed as follows:

$$p(n_{0:t}, \theta_{1:t} \mid x_{1:t}) = p(n_{0:t} \mid \theta_{1:t}, x_{1:t}) p(\theta_{1:t} \mid x_{1:t})$$

where, θ_t consists of the mean vector μ_t and covariance matrix Σ_t of clean speech signal and is drawn from the following Dirichlet process.

$$\mathrm{G} \sim DP(\mathrm{G}_0, \alpha)$$

$$\theta_t \sim \mathrm{G}$$

Then a clean speech is drawn from

$$s_t \sim f(\cdot \mid \theta_t)$$

A propability measure G_0 denotes, a Normal-inverse Wishart base distribution which is usually used when θ_t are a mean μ_t and a covariance Σ_t of gaussian law:

$$\mathrm{G}_0 = \mathcal{NIW}(\mu_0, \kappa_0, \nu_0, \Lambda_0)$$

with μ_0, κ_0, ν_0, Λ_0 the hyperparameters of the Normal-inverse Wishart. Sample from the Normal-inverse Wishart distribution is represented as follows:

$$\mu \mid \Sigma \sim \mathcal{N}(\mu_0, \frac{\Sigma}{\kappa_0})$$

$$\Sigma^{-1} \sim W(\nu_0, \Lambda_0^{-1})$$

where $\mathcal{N}$ is a gaussian distribution and W is the Wishart distribution. The parameters ν_0 and Λ_0 are the degree of freedom and the scale parameter of Wishart distribution respectively. μ_0 is the mean vector and κ_0 is also a scale parameter.

As $p(n_{0:t} \mid \theta_{1:t}, x_{1:t})$ can be computed using the EKF defined by Fujimoto *et al.* [15], we only need to estimate the probability $p(\theta_{1:t} \mid x_{1:t})$ using a particle method. At time *t*, it follows that $p(n_t, \theta_{1:t} \mid x_{1:t})$ is approximated through a set of *J* particles by the following empirical distribution

$$P_N(n_t, \theta_{1:t} \mid x_{1:t}) = \sum_{j=1}^{J} \tilde{\omega}_t^{(j)} p(n_t \mid \theta_{1:t}^{(j)}, x_{1:t})$$

with

$$p(n_t \mid \theta_{1:t}^{(j)}, x_{1:t}) \cong \mathcal{N}(\hat{n}_{t|t}(\theta_{1:t}^{(j)}), \Sigma_{n_{t|t}}^{(j)}(\theta_{1:t}^{(j)}))$$

The parameters $\hat{n}_{t|t}(\theta_{1:t}^{(j)})$ and $\Sigma_{n_{t|t}}^{(j)}(\theta_{1:t}^{(j)})$ are computed recursively for each particle *j* using the EKF. On the other hand, the posterior $p(\theta_{1:t}^{(j)} \mid x_{1:t})$ is proportional to $p(\theta_{1:t-1}^{(j)} \mid x_{1:t-1})$ as follows:

$$\begin{aligned} &p(\theta_{1:t}^{(j)} \mid x_{1:t}) \\ &\propto p(\theta_{1:t-1}^{(j)} \mid x_{1:t-1}) p(x_t \mid \theta_{1:t}^{(j)}, x_{1:t-1}) p(\theta_t^{(j)} \mid \theta_{1:t-1}^{(j)}) \end{aligned}$$

where

$$\begin{aligned} p(x_t \mid \theta_{1:t}^{(j)}, x_{1:t-1}) &= p(x_t \mid \theta_t^{(j)}, \theta_{1:t-1}^{(j)}, x_{1:t-1}) \\ &= \mathcal{N}(\hat{x}_t(\theta_{1:t}^{(j)}), \hat{\Sigma}_x^{(j)}(\theta_{1:t}^{(j)})) \end{aligned}$$

and

$$\hat{x}_t(\theta_{1:t}^{(j)}) = s_t^{(j)} + \log(I + \exp(n_t^{(j)} - s_t^{(j)}))$$

$$\hat{\Sigma}_x(\theta_{1:t}^{(j)}) = G_t^{(j)} \Sigma_{n_t}^{(j)} G_t^{(j)T} + \Sigma_{s,t}$$

$$G_t^{(j)} = \frac{\partial}{\partial n_t^{(j)}} \{ s_t^{(j)} + \log(I + \exp(n_t^{(j)} - s_t^{(j)})) \}$$

$$s_t^{(j)} \sim \mathcal{N}(\mu_t^{(j)}, \Sigma_t^{(j)})$$

Finally, sample weights are calculated using these estimates.

$$\tilde{\omega}_t^{(j)} \propto \omega_{t-1}^{(j)} \mathcal{N}(\hat{x}_t(\theta_{1:t}^{(j)}), \hat{\Sigma}_x(\theta_{1:t}^{(j)}))$$

because we chose the importance distribution as follows:

$$q(\theta_t^{(j)} \mid \theta_{1:t-1}^{(j)}, x_{1:t}) = p(\theta_t^{(j)} \mid \theta_{1:t-1}^{(j)})$$

$p(\theta_t^{(j)} \mid \theta_{t-1}^{(j)})$ is determined using the polya urn representation [7].

5.5 Introduction of Reverberation and Reflected Waves into the Proposed Model

In this section, we introduce reverberation and reflected waves into the proposed model. In real situations, speech signals are affected by reverberation and reflected waves. Also, speech signals decays when microphones are located far from the speakers. Let *h* denotes transfer characteristics in the log spectral domain and we assume a classical convolution in the time domain. We can get the following equation as an observation equation.

$$
\begin{aligned}
x_t &= s_t + h_t + \log(I + \exp(n_t - s_t - h_t)) + v_t \\
&= g(s_t, n_t, h_t) + v_t
\end{aligned}
$$

A transition equation for $h_t^{(j)}$ is defined as follows:

$$h_t^{(j)} = h_{t-1}^{(j)} + u_{t-1}^{(j)}$$

$$u_{t-1}^{(j)} \sim \mathcal{N}(0, \Sigma_u)$$

EKF is modified as follows:

$$
\begin{aligned}
n_{t|t-1}^{(j)} &= (1-\alpha_p) n_{t-1}^{(j)} + \alpha_p \hat{n}_{t-1} \\
&\quad + \alpha_p \beta_p (\mu_{n,t}^{(j)} - n_{t-1}^{(j)}) + w_{t-1}^{(j)}
\end{aligned}
$$

$$\Sigma_{n_{t|t-1}}^{(j)} = F_{t-1}^{(j)} \Sigma_{n_{t-1}}^{(j)} F_{t-1}^{(j)T} + \Sigma_w$$

$$h_{t|t-1}^{(j)} = h_{t-1}^{(j)} + u_{t-1}^{(j)}$$

$$\Sigma_{h_{t|t-1}}^{(j)} = \Sigma_{h_{t-1}}^{(j)} + u_{t-1}^{(j)}$$

$$K_{\eta,t}^{(j)} = \Sigma_{\eta_{t|t-1}}^{(j)} G_{\eta,t}^{(j)T} S_t^{(j)-1}$$

$$S_t^{(j)} = G_{n,t}^{(j)} \Sigma_{n_{t|t-1}}^{(j)} G_{n,t}^{(j)T} + G_{h,t}^{(j)} \Sigma_{h_{t|t-1}}^{(j)} G_{h,t}^{(j)T} + \Sigma_t$$

$$F_{t-1}^{(j)} = \alpha_p (-1 + \omega_{t-1}^{(j)} + \frac{\beta_p}{T_p})$$

$$G_{\eta,t}^{(j)} = \frac{\partial}{\partial \eta_{t|t-1}^{(j)}} g(\Psi^{(j)})$$

$$\hat{\eta}_t^{(j)} = \eta_{t|t-1}^{(j)} + K_{\eta,t}^{(j)} (x_t - g(\Psi^{(j)}))$$

$$\Sigma_{\eta_t}^{(j)} = \Sigma_{\eta_{t|t-1}}^{(j)} - K_{\eta,t}^{(j)} G_{\eta,t}^{(j)} \Sigma_{\eta_{t|t-1}}^{(j)}$$

where $\Psi^{(j)} = \{s_t^{(j)}, n_{t|t-1}^{(j)}, h_{t|t-1}^{(j)}\}$ and $\eta = [n \,|\, h]$. In order to estimate n_t, h_t and s_t, it is necessary to determine the probability $p(n_{0:t}, h_{0:t}, \theta_{1:t} \,|\, x_{1:t})$ which can be decomposed as follows:

$$
\begin{aligned}
& p(n_{0:t}, h_{0:t}, \theta_{1:t} \,|\, x_{1:t}) \\
& = p(n_{0:t} \,|\, \theta_{1:t}, x_{1:t}) p(h_{0:t} \,|\, \theta_{1:t}, x_{1:t}) p(\theta_{1:t} \,|\, x_{1:t})
\end{aligned}
$$

$p(n_{0:t} \,|\, \theta_{1:t}, x_{1:t})$ and $p(h_{0:t} \,|\, \theta_{1:t}, x_{1:t})$ are calculated by the EKF respectively and $p(\theta_{1:t} \,|\, x_{1:t})$ is calculated by the particle filtering which is shown in 5.4. The proposed method can finally be represented as the following algorithm.

```
Initialization
j = 1; ááá, J
        $n_0^{(j)} \sim \mathcal{N}(\mu_n, \Sigma_n)$
        $h_0^{(j)} = 0$
        $\omega_0^{(j)} = \frac{1}{J}$
end
t = 1; ááá, T
        calculate $\mu_0, \Lambda_0$
```

```
    j = 1; ááą J
        if t == 1
            $\theta_t^{(j)} \sim \mathcal{NIW}(\mu_0, \kappa_0, \nu_0, \Lambda_0)$
        else
            $\theta_t^{(j)} \sim p(\theta_t^{(j)} \mid \theta_{t-1}^{(j)})$
        end
          $s_t^{(j)} \sim \mathcal{N}(\mu_t^{(j)}, \Sigma_t^{(j)})$        $\theta_t^{(j)} = \{\mu_t^{(j)}, \Sigma_t^{(j)}\}$
        switching dynamical system [16]
        EKF
        $[\hat{x}_t(\theta_{1:t}^{(j)}), \hat{\Sigma}_x(\theta_{1:t}^{(j)}), n_t^{(j)}, \Sigma_{n_t}^{(j)}, h_t^{(j)}, \Sigma_{h_t}^{(j)}]$
        $= EKF(n_{t-1}^{(j)}, \Sigma_{n_{t-1}}^{(j)}, \theta_t^{(j)}, h_{t-1}^{(j)}, \Sigma_{h_{t-1}}^{(j)}, x_t)$
        calculate sample weights
        $\widetilde{\omega}_t^{(j)} \propto \omega_{t-1}^{(j)} \mathcal{N}(\hat{x}_t(\theta_{1:t}^{(j)}), \hat{\Sigma}_x(\theta_{1:t}^{(j)}))$
    end
    $\Sigma_{j=1}^{J} \widetilde{\omega}_t^{(j)} = 1$
    compute    $N_{eff} = \{\Sigma_{j=1}^{J} (\widetilde{\omega}_t^{(j)})^2\}^{-1}$
    if $N_{eff} \le \varepsilon$, resample the particles and $\omega_t^{(j)} = \frac{1}{J}$
    $\hat{n}_t = \Sigma_{j=1}^{J} \omega_t^{(j)} n_t^{(j)}$
    $\hat{s}_t = \Sigma_{j=1}^{J} \omega_t^{(j)} s_t^{(j)}$
end
```

6. Simulations

6.1 Simulation Setup

We compare three processing schemes: first one is a method proposed by Fujimoto *et al.* [16] where Vector Taylor Series method is not employed (conventional), second one is the proposed method without considering transfer characteristics and third one is the proposed method with considering transfer characteristics. Three types of data set are made for evaluations. First one is clean speeches recorded in a sound proof chamber, second one is noisy speeches which are artificially generated by adding three types of noises and third one is noisy reverberant speeches which are artificially generated by convolving transfer characteristics with the noisy speeches. Noise data are taken from “Sound Scene Database in Real Acoustical Environment” [22]. We employ white noise, particle noise and shaver noise. Then, these noises are artificially added to clean speeches with SNRs from 0 to 9dB. Transfer characteristics are simulated using the image method [1]. Reverberation time of the simulated data is about 500ms. 100 utterances uttered by four males and two females are used for this evaluation. The contents of the utterances are TV controlling commands, e.g “volume up”, “turn off” and so on. The total number of evaluation data for each SNR is 3,600 short phrases.

GMM with 256 mixture distributions is trained using 500 utterances uttered by 3 males and 2 females.

An acoustic model for speech recognition is developed using the Acoustical Society of Japan (ASJ) continuous speech corpus [21]. The training data are about 30,000 sentences uttered by 150 males and 150 females. The feature parameters for the acoustic model is composed of 39 Mel Frequency Cepstral Coefficients (MFCCs) [10] with 13 MFCCs (with zero-th MFCC) and their first and second order derivatives. At the feature extraction stage, Cepstral Mean Subtraction (CMS) [4] is applied to each sentence.

Parameters for the particle filtering are as follows: w_t is set to $\Sigma_w = 0.1$, u_t is set to $\Sigma_u = 0.0001$ and z_t is set to $\Sigma_z = 1$. The number of particles is 100. Parameters for the Polyak averaging and feedback have four states respectively, e.g. α_p = {0.05, 0.1, 0.15, 0.2}, β_p = {0.5, 1.0.1.5, 2.0} and T_p = {5, 10, 15, 20}. Moreover, A parameter for the switching dynamical system is γ = 0.5 [16].

The parameter α for DPM is different according to the length of utterance. Because, as the result of a preliminary experiment, it is clear that short phrases can be recognized even if α is a low number, while in order to recognize the long phrases, it is necessary that α is a high number.

We have no *a priori* information on the speech signal distribution. As we do not know which value is better for hyperparameters, a mechanism for estimating hyperparamters is introduced. This estimation bases on the difference between the received signal and the received signal estimated using the estimated clean signal at *t*-1 and the estimated noise signal at *t*. That is to say, at the time *t*, the clean signal is estimated roughly as follows:

$$\tilde{s}_t^{(j)} = s_{t-1}^{(j)} + \Delta s^{(j)} + z_t^{(j)}$$

$$\Delta s^{(j)} = x_t - (s_{t-1}^{(j)} + \log(1 + \exp(\hat{n}_t^{(j)} - s_{t-1}^{(j)})))$$

where $\hat{n}_t^{(j)}$ is obtained from the Polyak averaging [7] and $z_{t-1}^{(j)} \sim \mathcal{N}(0, \Sigma_z)$. Then, the mean vector and covariance matrix of these particles are calculated and we regard these values as μ_0 and Λ_0 of hyperparameters.

$$\mu_0 = \frac{1}{J}\sum_{j=1}^{J} \tilde{s}_t^{(j)}$$

$$\Lambda_0 = \sqrt{\frac{1}{J}\sum_{j=1}^{J} (\tilde{s}_t^{(j)} - \mu_t)^2}$$

Then κ_0 =1 and ν_0 =500.

6.2 Results

Firstly, the noise and clean speech estimation results are shown. Figure 1 shows one example of the noise and speech tracking results by the proposed method.

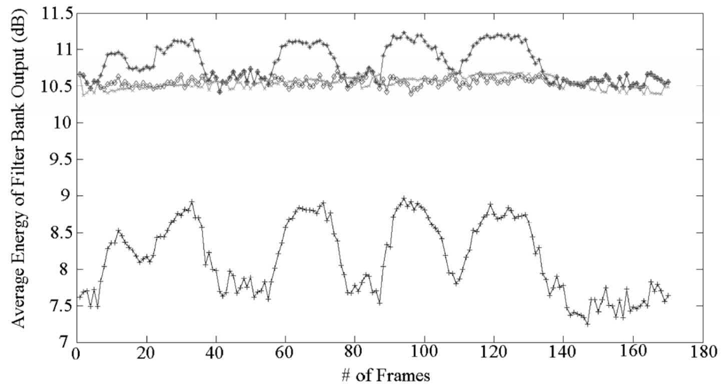

Figure 1. Tracking result of the proposed method (in case where a noise signal is a white noise and SNR is 3dB) *: received (observed signal), ◊ : true noise signal, × : estimated noise signal, +: estimated clean signal

The abscissa is the number of frame and the ordinate is the average energy of filter bank output in the log spectral domain. It is clear that the proposed method can track the noise sequence in case SNR is 3dB. Figure 2 shows one example of the difference between the true noise sequence and the estimated noise sequence.

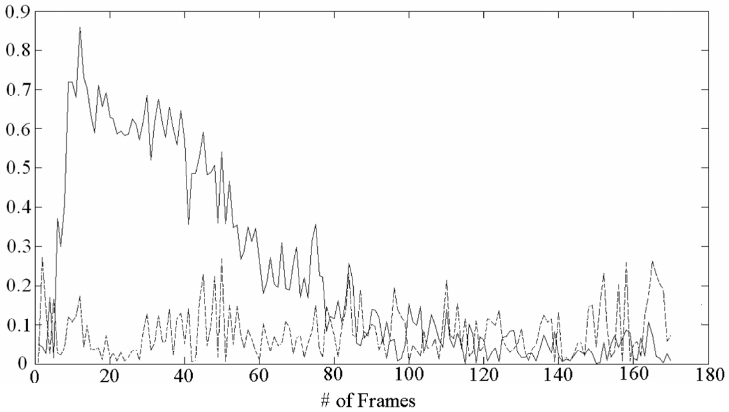

Figure 2. Difference average filter bank output between an estimated noise sequence and a true noise sequence in case where a noise signal is a white noise and SNR is 3dB (solid line: the method using GMM, dotted line: the proposed method)

You can see that the conventional method fails to track a noise sequence at the start of phrase. On the other hand, the proposed method works more badly than the conventional method at the end of phrase. The reason is that the conventional method using GMM cannot estimate the sudden change of the noise sequence. However, the proposed method using DPM permits the flexible noise sequence estimation.

Secondly, the speech recognition rates are compared. Evaluations are performed using speech recognition decoder "Julian" [23]. Clean speeches are recorded in a sound proof chamber using a close contact microphone. Table 1 shows the condition of acoustical analysis.

Table 1: Condition of acoustical analysis

sampling rate	16000 samples/sec
frame size	512
window size	400
frame shift	160
feature parameter	39 dimensional mfccs (12mfccs+C0+12 Δ mfccs+ Δ C0 +12 Δ Δ mfccs+ Δ Δ C0)
cepstrum coefficient	24 dimension

Tables 2 and 3 show speech recognition rates in case we did not considered the effect of the reverberation and the reflected waves and we considered respectively.

Table 2: Speech Recognition Rate for Noisy Data (%)

	white			shaver			particle		
	no_processing	proposed	conventional	no_processing	proposed	conventional	no_processing	proposed	conventional
0dB	2.8	19.8	3.0	7.8	31.3	7.0	8.0	26.7	10.2
3dB	15.3	55.7	11.2	36.7	56.7	17.0	30.8	50.3	21.8
6dB	50.3	78.2	33.0	63.3	71.7	37.7	61.5	69.3	39.0
9dB	79.8	88.0	52.8	79.8	78.2	52.2	83.8	77.7	53.5

Table 3: Speech Recognition Rate for Noisy and Reverberant Data (%)

	white			shaver			particle		
	no_processing	proposed	conventional	no_processing	proposed	conventional	no_processing	proposed	conventional
0dB	2.2	12.8	2.0	3.0	14.0	3.8	2.0	13.2	3.7
3dB	8.3	38.0	8.8	21.8	33.8	9.8	14.5	27.8	9.3
6dB	33.2	56.8	17.3	46.8	47.3	20.0	37.0	44.0	19.8
9dB	65.7	68.0	36.8	66.0	58.0	35.8	62.7	54.8	35.7

In these tables, the speech recognition rate for three types of noise data (white noise, shaver noise, particle noise) are shown. Moreover, for each noise data, there are the speech recognition rates of three processing schemes (no processing, conventional method, proposed method). From this table, it can be found that speech recognition rates are improved using the proposed method in case where the SNRs are 0, 3 and 6dB. On the other hand, in case the SNR is 9dB, speech recognition rates are degraded except for the case of white noise. The reason why this degradation of speech recognition rate is that the noise tracking performance degrades as shown in Fig. 3.

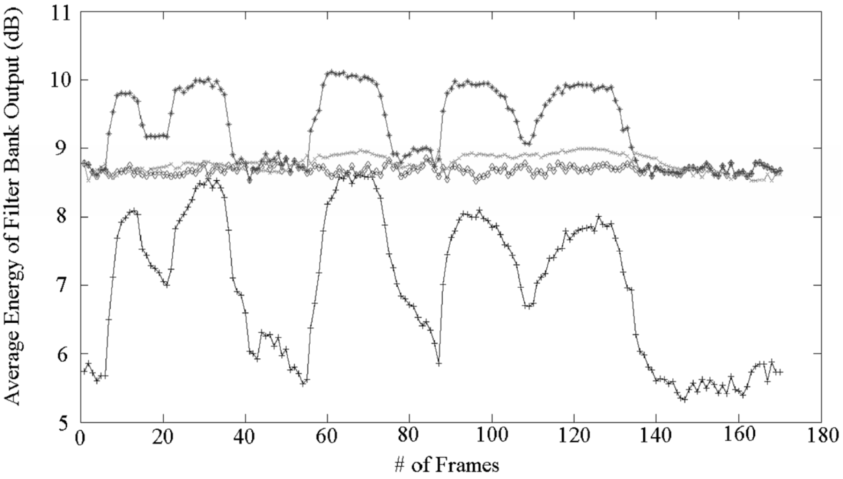

Figure 3. Tracking result of the proposed method (in case where a noise signal is a white noise and SNR is 9dB) *: received (observed signal), ◊ : true noise signal, × : estimated noise signal, +: estimated clean signal

In case SNR is 9dB, noise signal power is small and the fluctuation of noise signal is also small, while the fluctuation of clean speech is large. Nevertheless, the covariance of $w_{t-1}^{(j)}$ in eq. (11) is too large and the covariance of $z_t^{(j)}$ in eq. (17) is too small. As the result, estimated noise sequence becomes larger than the true noise sequence, on the other hand estimated clean speech sequence becomes smaller than the true clean speech sequence.

The speech recognition rate by the conventional method is lower than even that with no processing. The reason is that the time allocated to the GMM learning is not enough long.

7. Conclusion

In this paper, we proposed a method for modeling the clean speech distribution using DPM and noise sequence using particle filtering. Our proposed method realizes better noise estimation accuracy than the method using inaccurate GMM. In the evaluation using speech recognition, our proposed method can improve the speech recognition rate in the SNRs 0dB, 3dB 6dB except for the White noise. On the other hand, in case of high SNR, tracking performance degrades because of the problem of parameter setting. So, our future work is the estimation of the covariance of $w_{t-1}^{(j)}$ in eq. (11) and the covariance of $z_t^{(j)}$ in eq. (17).

REFERENCES

1. JONT, B., ALLEN and DAVID A. BERKLEY, **Image Method for Efficiently Simulating Small-Room Acoustics**, Journal of ASA, Vol. 65, No. 4, Apr., 1979, pp. 943–950.

2. ANTONIAK, C. E., **Mixtures of Dirichlet Processes with Applications to Nonparametric Problems**, Annals of Statistics, 2, 1974, pp. 1152–1174.

3. ARULAMPALAM, M. S,. 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.

4. ATAL, B. S., **EffecTiveness of Linear Prediction Characteristics of the Speech Wave for Automatic Speaker Identification and Verification**, J. Acoust. Soc. Am. 55 (6), pp. 1304-1312, 1974.

5. BLACKWELL, D. and J. B. MACQUEEN, **Ferguson Distributions Via Polya Urn Schemes**, Annals of Statistics, 1:353-355, 1973.

6. BOLL, S. F., **Suppression of Acoustic Noise in Speech Using Spectral Subtraction**, IEEE Trans. ASSP, Vol. 27, No. 2, pp. 113-120, 1979.

7. CARON, F., M. DAVY, A. DOUCET, E. DUFLOS, P. VANHEEGHE, **Bayesian Inference for Dynamic Models with Dirichlet Process Mixtures**, International Conference on Information Fusion (FUSION'06), Florence, Italia, July 10-13, 2006.

8. CARON, F., **Inference bayesienne pour la d'etermination et la selection de modeles stochastiques,** Doctral thesis of ecole centrale de Lille.

9. COMON, P., **Independent Component Analysis: A New Concept?** Signal Proc., Vol. 36, 1994, pp. 287-314.

10. DAVIS, S. B. and P. MERMELSTEIN, **Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences,** IEEE Trans. Acoustics, Speech, and Signal Processing, Vol. 28, No. 4, 1980, pp. 357-366.

11. DOUCET, A., N. DE FREITAS and N. GORDON, **Sequential Monte Carlo Methods in Practice**, Springer-Verlag, 2001.

12. ESCOBAR, M. D. and M. WEST, **Bayesian Density Estimation and Inference Using Mixtures**, Journal of the American Statistical Association, Vol. 90, No. 430, 1995.

13. FERGUSON, T. S., **A Bayesian Analysis of Some Nonparametric Problems**, Annals of Statistics 1, 1973, pp. 209-230.

14. FLANAGAN, J. L., J. D. JOHNSTON, R. ZAHN and G.W. ELKO, **Computer-Steered Microphone Arrays for Sound Transduction in Large Rooms**, JASA, Vol. 78, Nov. 1985, pp. 1508-1518.

15. FUJIMOTO, M. and S. NAKAMURA, **Particle Filtering and Polyak Averaging-Based Non-Stationary Noise Tracking for ASR in Noise**, Proc. ASRU '05, Nov. 2005, pp. 337-342.

16. FUJIMOTO, M. and S. NAKAMURA, **Sequential Nonstationary Noise Tracking Using Particle Filtering with Switching Dynamical System**, Proc. ICASSP 2006, May 2006, pp. 769-772.

17. GAMERMAN, D. and H. F. LOPES, **Markov Chain Monte Carlo**, Chapman & Hall/CRC, 2006.

18. GRIFFTH, L. J. and C. W. JIM, **An Alternative Approach to Linearly Constrained Adaptive Beamforming**, IEEE Trans. Antennas Propagation, Vol. 30, No. 1, 1982, pp.27-34.

19. KOTTAS, A., **Dirichlet Process Mixtures of Beta Distributions, with Applications to Density and Intensity Estimation**, Proceedings of the Workshop on Learning with Nonparametric Bayesian Methods, 23rd ICML, 2006.

20. J. C. SEGURA, A. DE LA TORRE, M. C. BENITEZ, and A. M. PEINADO, **Model-Basd Compensation of the Additive Noise for Continuous Speech Recognition. Experiments Using AURORA II Database and Tasks**, Proc. EuroSpeech '01, Vol. I, Sept. 2001, pp. 221-224.

21. http://www.mibel.cs.tsukuba.ac.jp/jnas/instruct.html

22. http://tosa.mri.co.jp/sounddb/indexe.htm

23. http://julius.sourceforge.jp/