AN ALGORITHM FOR SPEECH PARAMETER GENERATION FROM CONTINUOUS MIXTURE HMMS WITH DYNAMIC FEATURES

Keiichi Tokuda† *Takashi Masuko*‡ *Tetsuya Yamada*‡ *Takao Kobayashi*‡ *Satoshi Imai*‡

†Department of Electrical and Electronic Engineering, Tokyo Institute of Technology, Tokyo, 152 Japan

‡Precision and Intelligence Laboratory, Tokyo Institute of Technology, Yokohama, 226 Japan

tokuda@ss.titech.ac.jp, {masuko, tetsuya, tkobayas, imai}@pi.titech.ac.jp

ABSTRACT

This paper proposes an algorithm for speech parameter generation from continuous mixture HMMs which include dynamic features, i.e., delta and delta-delta parameters of speech. We show that the parameter generation from HMMs using the dynamic features results in searching for the optimal state sequence and solving a set of linear equations for each possible state sequence. To solve the problem, we derive a fast algorithm on the analogy of the RLS algorithm for adaptive filtering. We show that the generated speech parameter vectors reflect not only the means of static and dynamic feature vectors but also the covariances of those. An example presenting effectiveness of the proposed algorithm in speech synthesis is given.

1. INTRODUCTION

The hidden Markov models (HMMs) can model sequences of speech spectra with well-defined algorithms, and have successfully been applied to speech recognition systems. From these facts, we surmise that HMMs are also useful for speech synthesis. Actually, some attempts have been made in this context [1], [2], [3], [4]. If we can synthesize speech from HMMs, it will be feasible to synthesize speech with various voice quality by applying fast speaker adaptation techniques used in HMM-based speech recognition, and synthesis units can be selected automatically based on the model clustering and splitting methods. In addition, it is expected that the speech synthesis method is applicable to speech enhancement, vector quantization of speech spectral parameters [5], voice conversion, etc.

From this point of view, we have proposed an algorithm [6] for speech parameter generation from single-mixture HMMs with dynamic feature, i.e., delta parameter of speech. In this paper, we propose an algorithm for continuous mixture HMMs which include delta and delta-delta parameters of speech. We show that the dynamic features play an important role in speech parameter generation from HMMs, that is, by using dynamic features, the generated speech parameter vectors reflect not only the means of static and dynamic feature vectors but also the covariances of those, whereas, without dynamic features, the generated speech parameter sequence reflects only the means of static feature vectors.

2. PROBLEM

For a given continuous mixture HMM λ, we consider mixture components to be sub-states in a similar manner of the Viterbi algorithm, and maximize $P[\mathbf{Q}, \mathbf{O} \,|\, \lambda]$ with respect to speech parameter vector sequence $\mathbf{O} = [\mathbf{o}_1', \mathbf{o}_2', \ldots, \mathbf{o}_T']'$ and sub-state sequence $\mathbf{Q} = \{(q_1, i_1), (q_2, i_2), \ldots, (q_T, i_T)\}$, where (q, i) indicates the i-th mixture of state q.

In this paper, we assume that the speech parameter vector $\mathbf{o}_t$ at frame t consists of the static feature vector $\mathbf{c}_t = [c_t(1), c_t(2), \ldots, c_t(M)]'$ (e.g., cepstral coefficients) and the dynamic feature vectors $\Delta\mathbf{c}_t$, $\Delta^2\mathbf{c}_t$ (e.g., delta and delta-delta cepstral coefficients, respectively), that is, $\mathbf{o}_t = [\mathbf{c}_t', \Delta\mathbf{c}_t', \Delta^2\mathbf{c}_t']'$. By setting $\Delta^{(0)}\mathbf{c}_t = \mathbf{c}_t$, $\Delta^{(1)}\mathbf{c}_t = \Delta\mathbf{c}_t$, $\Delta^{(2)}\mathbf{c}_t = \Delta^2\mathbf{c}_t$, we can define

$$\Delta^{(n)}\mathbf{c}_t = \sum_{i=-L^{(n)}}^{L^{(n)}} w^{(n)}(i)\,\mathbf{c}_{t+i}, \quad n = 0, 1, 2 \qquad (1)$$

where $L^{(0)} = 0$, $w^{(0)}(0) = 1$.

To control temporal structure appropriately, HMMs should incorporate state duration. Heuristic duration densities can also be used for this purpose. Thus, the logarithm of $P[\mathbf{Q}, \mathbf{O} \,|\, \lambda]$ can be written as

$$\begin{aligned}\log P[\mathbf{Q}, \mathbf{O} \,|\, \lambda] = \alpha \sum_{k=1}^{K} \log p_{q_k}(d_{q_k}) + \sum_{t=1}^{T} \log c_{q_t, i_t} \\ -\frac{1}{2}(\mathbf{O} - \boldsymbol{\mu})'\mathbf{U}^{-1}(\mathbf{O} - \boldsymbol{\mu}) \\ -\frac{1}{2}\log|\,\mathbf{U}\,| - \frac{3MT}{2}\log 2\pi \qquad (2)\end{aligned}$$

where

$$\boldsymbol{\mu} = \left[\boldsymbol{\mu}_{q_1, i_1}', \boldsymbol{\mu}_{q_2, i_2}', \ldots, \boldsymbol{\mu}_{q_T, i_T}'\right]' \qquad (3)$$

$$\mathbf{U} = \mathrm{diag}\left[\mathbf{U}_{q_1, i_1}, \mathbf{U}_{q_2, i_2}, \ldots, \mathbf{U}_{q_T, i_T}\right], \qquad (4)$$

and $c_{q,i}$, $\boldsymbol{\mu}_{q_t, i_t}$ and $\mathbf{U}_{q_t, i_t}$ are the mixture weight, the $3M \times 1$ mean vector and the $3M \times 3M$ covariance matrix associated with i_t-th mixture of state

q_t, respectively. We assume that the total number of states which have been visited during T frames is K, and $p_{q_k}(d_{q_k})$ is the probability of d_{q_k} consecutive observations in state q_k.

From (2), it is evident that $P[\mathbf{Q},\, \mathbf{O}\,|\,\lambda]$ is maximized when $\mathbf{O} = \boldsymbol{\mu}$, that is, the speech parameter vector sequence becomes a sequence of the mean vectors independently of the covariance $\mathbf{U}$. To avoid this, we should maximize $P[\mathbf{Q},\, \mathbf{O}\,|\,\lambda]$ with respect to $\mathbf{Q}$ and $\mathbf{O}$ under the constraints (1) on $\mathbf{O}$, equivalently maximize $P[\mathbf{Q},\, \mathbf{O}\,|\,\lambda]$ with respect to $\mathbf{Q}$ and $\mathbf{c} = [\mathbf{c}_1',\, \mathbf{c}_2',\, \ldots,\, \mathbf{c}_T']'$ under the constraints (1). In contrast to the Viterbi algorithm, it is noted that the dynamic programming methods cannot be used since we have to determine $\mathbf{Q}$ and $\mathbf{c}$ simultaneously.

3. SOLUTION FOR THE PROBLEM

First let us consider maximizing $P[\mathbf{Q},\, \mathbf{O}\,|\,\lambda]$ with respect to $\mathbf{c}$ for a fixed sub-state sequence $\mathbf{Q}$. By using (1), $\log P[\mathbf{Q},\, \mathbf{O}\,|\,\lambda]$ can be rewritten as

$$\log P[\mathbf{Q},\, \mathbf{O}\,|\,\lambda] = \alpha \sum_{k=1}^{K} \log p_k(d_k) + \sum_{t=1}^{T} \log c_{q_t,\, i_t} - \frac{1}{2}\varepsilon(\mathbf{c}) - \frac{1}{2}\log|\,\mathbf{U}\,| - \frac{3MT}{2}\log 2\pi \qquad (5)$$

where

$$\varepsilon(\mathbf{c}) = (\mathbf{W}\mathbf{c} - \boldsymbol{\mu})'\mathbf{U}^{-1}(\mathbf{W}\mathbf{c} - \boldsymbol{\mu}) \qquad (6)$$

and

$$\mathbf{W} = [\mathbf{w}_1,\, \mathbf{w}_2,\, \ldots,\, \mathbf{w}_T]' \qquad (7)$$

$$\mathbf{w}_t = [\mathbf{w}_t^{(0)},\, \mathbf{w}_t^{(1)},\, \mathbf{w}_t^{(2)}] \qquad (8)$$

$$\mathbf{w}_t^{(n)} = [\underset{1\text{st}}{\mathbf{0}_{M\times M}},\, \ldots,\, \mathbf{0}_{M\times M},\, \underset{(t-L^{(n)})\text{-th}}{w^{(n)}(-L^{(n)})I_{M\times M}},\, \ldots,\, \underset{t\text{-th}}{w^{(n)}(0)I_{M\times M}},\, \ldots,\, \underset{(t+L^{(n)})\text{-th}}{w^{(n)}(L^{(n)})I_{M\times M}},\, \underset{T\text{-th}}{\mathbf{0}_{M\times M}},\, \ldots,\, \mathbf{0}_{M\times M}]', \quad n = 0,\, 1,\, 2, \qquad (9)$$

where $\mathbf{0}_{M\times M}$ and $\mathbf{I}_{M\times M}$ denote the $M\times M$ zero matrix and the $M\times M$ identity matrix, respectively. We assume that $\mathbf{c}_t = \mathbf{0}_M$, $t < 1$, $T < t$, where $\mathbf{0}_M$ denotes the $M\times 1$ zero vector. Thus, by setting $\partial \log P[\mathbf{Q},\, \mathbf{O}\,|\,\lambda]/\partial\mathbf{c} = \mathbf{0}_{TM}$, we obtain a set of equations

$$\mathbf{R}\mathbf{c} = \mathbf{r} \qquad (10)$$

where

$$\mathbf{R} = \mathbf{W}'\mathbf{U}^{-1}\mathbf{W} \qquad (11)$$

$$\mathbf{r} = \mathbf{W}'\mathbf{U}^{-1}\boldsymbol{\mu}. \qquad (12)$$

For direct solution of (10), we need $O(T^3M^3)$ operations because $\mathbf{R}$ is a $TM\times TM$ matrix. When $\mathbf{U}_{q,\,i}$ is diagonal, it becomes $O(T^3M)$ since each element of speech parameter vector can be calculated independently. To obtain $\mathbf{Q}$ and $\mathbf{c}$ which maximize $P[\mathbf{Q},\, \mathbf{O}\,|\,\lambda]$, we have to solve (10) for every possible sub-state sequence. Fortunately, by using special properties of (10), we can derive a fast algorithm for determination of $\mathbf{Q}$ and $\mathbf{c}$ as follows.

Assuming that a frame t belongs to i_t-th mixture of state q_t, let us consider replacing $(q_t,\, i_t)$ of the frame t with $(\hat{q}_t,\, \hat{\imath}_t)$. The corresponding set of equations can be written as

$$\hat{\mathbf{R}}\hat{\mathbf{c}} = \hat{\mathbf{r}} \qquad (13)$$

where

$$\hat{\mathbf{R}} = \mathbf{R} + \mathbf{w}_t\,\mathbf{D}\,\mathbf{w}_t' \qquad (14)$$

$$\hat{\mathbf{r}} = \mathbf{r} + \mathbf{w}_t\,\mathbf{d} \qquad (15)$$

$$\mathbf{D} = \mathbf{U}^{-1}_{\hat{q}_t,\,\hat{\imath}_t} - \mathbf{U}^{-1}_{q_t,\,i_t} \qquad (16)$$

$$\mathbf{d} = \mathbf{U}^{-1}_{\hat{q}_t,\,\hat{\imath}_t}\,\boldsymbol{\mu}_{\hat{q}_t,\,\hat{\imath}_t} - \mathbf{U}^{-1}_{q_t,\,i_t}\,\boldsymbol{\mu}_{q_t,\,i_t}. \qquad (17)$$

It can be seen that the relation between $\mathbf{R}$ and $\hat{\mathbf{R}}$ is similar to the time update property of the set of equations for the RLS adaptive filtering [7], that is, the rank of $\mathbf{w}_t\,\mathbf{D}\,\mathbf{w}_t'$ is $3M$ whereas the rank of $\mathbf{R}$ is TM. Consequently, on the analogy of the derivation of the standard RLS algorithm, i.e., the application of the matrix inversion lemma, we can derive a fast algorithm for obtaining $\hat{\mathbf{c}}$ from $\mathbf{c}$ recursively.

The algorithm is shown in Table 1. It is noted that $\mathbf{P} = \mathbf{R}^{-1}$. Since almost each element of $\mathbf{w}_t$ equals zero, (T.6) mainly has effect on the computational complexity, which is $O(T^2M^3)$. When $\mathbf{U}_{q,\,i}$ is diagonal, it is reduced to $O(T^2M)$. Furthermore, if we assume that the mean and covariance at a frame t have the influence only on the speech parameter vectors at S neighborhood frames, the computational complexity is reduced to $O(S^2M^3)$ (in case of diagonal covariance, $O(S^2M)$). Empirically, 30 is sufficient value for S.

Table 1: Algorithm to replace the sub-state $(q_t,\, i_t)$ of a frame t with $(\hat{q}_t,\, \hat{\imath}_t)$.

Substitute $\hat{\mathbf{c}}$, $\hat{\mathbf{P}}$ and $\hat{\varepsilon}$ obtained by the previous iteration to $\mathbf{c}$, $\mathbf{P}$ and ε respectively, and calculate

$$\boldsymbol{\pi} = \mathbf{P}\,\mathbf{w}_t \qquad (\text{T.1})$$

$$\boldsymbol{\nu} = \mathbf{w}_t'\,\boldsymbol{\pi} \qquad (\text{T.2})$$

$$\mathbf{k} = \boldsymbol{\pi}\left\{\mathbf{I}_{3M} + \left(\mathbf{U}^{-1}_{\hat{q}_t,\,\hat{\imath}_t} - \mathbf{U}^{-1}_{q_t,\,i_t}\right)\boldsymbol{\nu}\right\}^{-1} \qquad (\text{T.3})$$

$$\hat{\mathbf{c}} = \mathbf{c} + \mathbf{k}\{\mathbf{U}^{-1}_{\hat{q}_t,\,\hat{\imath}_t}\left(\boldsymbol{\mu}_{\hat{q}_t,\,\hat{\imath}_t} - \mathbf{w}_t'\,\mathbf{c}\right) - \mathbf{U}^{-1}_{q_t,\,i_t}\left(\boldsymbol{\mu}_{q_t,\,i_t} - \mathbf{w}_t'\,\mathbf{c}\right)\} \qquad (\text{T.4})$$

$$\hat{\varepsilon} = \varepsilon + \left(\boldsymbol{\mu}_{\hat{q}_t,\,\hat{\imath}_t} - \mathbf{w}_t'\,\hat{\mathbf{c}}\right)'\mathbf{U}^{-1}_{\hat{q}_t,\,\hat{\imath}_t}\left(\boldsymbol{\mu}_{\hat{q}_t,\,\hat{\imath}_t} - \mathbf{w}_t'\,\mathbf{c}\right) - \left(\boldsymbol{\mu}_{q_t,\,i_t} - \mathbf{w}_t'\,\hat{\mathbf{c}}\right)'\mathbf{U}^{-1}_{q_t,\,i_t}\left(\boldsymbol{\mu}_{q_t,\,i_t} - \mathbf{w}_t'\,\mathbf{c}\right) \qquad (\text{T.5})$$

$$\hat{\mathbf{P}} = \mathbf{P} - \mathbf{k}\left(\mathbf{U}^{-1}_{\hat{q}_t,\,\hat{\imath}_t} - \mathbf{U}^{-1}_{q_t,\,i_t}\right)\boldsymbol{\pi} \qquad (\text{T.6})$$

By using the recursive algorithm, we can search for the optimum sub-state sequence keeping $\mathbf{c}$ optimal in the sense that $P[\mathbf{Q}, \mathbf{O} \mid \lambda]$ is maximized with respect to $\mathbf{c}$. For each sub-state sequence, we can use the recursive algorithm instead of solving (10) directly. Therefore the total computational complexity is significantly reduced.

There are many strategies to choose the frame t whose sub-state is replaced. A procedure obtained with a strategy which we chose can be summarized as follows:

1. Initialization

(a) Determine the initial sub-state sequence $\mathbf{Q}$.
(b) For the initial sub-state sequence, obtain $\mathbf{c}$, ε and $\mathbf{P}$.

2. Iteration

(a) For $t = 1, 2, \ldots, T$,
  (i) Calculate (T.1), (T.2).
  (ii) For each possible sub-state of the frame t, calculate (T.3)-(T.5) and obtain $\log P[\mathbf{Q}, \mathbf{O} \mid \lambda]$ by (5).
  (iii) Choose the best sub-state in the sense that $\log P[\mathbf{Q}, \mathbf{O} \mid \lambda]$ is most increased by the sub-state replacement.
(b) Choose the best frame in the sense that $\log P[\mathbf{Q}, \mathbf{O} \mid \lambda]$ is most increased by the sub-state replacement.
(c) If $\log P[\mathbf{Q}, \mathbf{O} \mid \lambda]$ cannot be increased by the sub-state replacement at the best frame, stop iterating.
(d) Replace the sub-state of the best frame by calculating (T.1)-(T.6), and obtain $\hat{\mathbf{c}}$, $\hat{\mathbf{P}}$ and $\hat{\varepsilon}$.
(e) Go to 2(a)

Since the above procedure does not search every possible sub-state sequence, the initial sub-state sequence should be close to the optimum sub-state sequence in order to obtain an optimal or sub-optimal solution without a large number of iterations of the proposed algorithm. A reasonable way is shown in the following.

By using the Viterbi algorithm, determine state sequence $\mathbf{q} = \{q_1, q_2, \ldots, q_T\}$ in such a way that

$$\log P[\mathbf{q} \mid \lambda] = \sum_{k=1}^{K} \log p_{q_k}(d_{q_k}) \tag{18}$$

is maximized with respect to $\mathbf{q}$. Determine mixture sequence $\mathbf{i} = \{i_1, i_2, \ldots, i_T\}$ in such a way that $\log c_{q_t, i_t} - (1/2) \log |\mathbf{U}_{q_t, i_t}|$ is maximized with respect to i_t. Assuming imaginary sub-states whose means and covariances are given by

$$\overline{\boldsymbol{\mu}}_{q_t, i_t} = [\boldsymbol{\mu}'^{(0)}_{q_t, i_t}, \mathbf{0}'_M, \mathbf{0}'_M]' \tag{19}$$

$$\overline{\mathbf{U}}^{-1}_{q_t, i_t} = \begin{bmatrix} \left(\mathbf{U}^{(0)}_{q_t, i_t}\right)^{-1} & \mathbf{0}_{M\times M} & \mathbf{0}_{M\times M} \\ \mathbf{0}_{M\times M} & \mathbf{0}_{M\times M} & \mathbf{0}_{M\times M} \\ \mathbf{0}_{M\times M} & \mathbf{0}_{M\times M} & \mathbf{0}_{M\times M} \end{bmatrix} \tag{20}$$

where $\boldsymbol{\mu}^{(0)}_{q_t, i_t}$ and $\mathbf{U}^{(0)}_{q_t, i_t}$ are the $M \times 1$ mean vector and the $M \times M$ covariance matrix of static feature vector $\mathbf{c}_t$ associated with i_t-th mixture of state q_t, respectively, we obtain

$$\overline{\mathbf{c}} = \left[\boldsymbol{\mu}'^{(0)}_{q_1, i_1}, \boldsymbol{\mu}'^{(0)}_{q_2, i_2}, \ldots, \boldsymbol{\mu}'^{(0)}_{q_T, i_T}\right]' \tag{21}$$

$$\overline{\mathbf{P}} = \mathrm{diag}\left[\mathbf{U}^{(0)}_{q_1, i_1}, \mathbf{U}^{(0)}_{q_2, i_2}, \ldots, \mathbf{U}^{(0)}_{q_T, i_T}\right] \tag{22}$$

and $\overline{\varepsilon} = 0$. By putting the values of $\overline{\boldsymbol{\mu}}_{q_t, i_t}$ and $\overline{\mathbf{U}}_{q_t, i_t}$ back with the original values of $\boldsymbol{\mu}_{q_t, i_t}$ and $\mathbf{U}_{q_t, i_t}$ for $t = 1, 2, \ldots, T$ using the algorithm T times, we can obtain $\mathbf{c}$, $\mathbf{P}$ and ε for the initial sub-state sequence.

4. SPEECH SYNTHESIS BASED ON HMM

Although we suppose that mel-cepstrum is used as speech parameter, the LPC-derived mel-cepstrum is not proper for synthesizing speech by the following reasons. Because of the truncation of mel-cepstral coefficients, it does not represent the original spectrum obtained by the LPC analysis, and we cannot obtain stable LPC synthesis filter from the LPC-derived mel-cepstrum. Similarly, MFCCs (Mel Frequency Cepstrum Coefficients), which are calculated by FFT-based band pass filtering, are not suitable for synthesizing speech. We have proposed a speech analysis method [8] and a speech synthesis method [9] in which speech spectrum is represented by mel-cepstrum consistently. In the mel-cepstral analysis method, speech spectrum is modeled by M mel-cepstral coefficients, and a spectral criterion is minimized with respect to the mel-cepstral coefficients. By using the MLSA (Mel Log Spectral Approximation) filter [9], [8], we can synthesize speech from the mel-cepstral coefficients, directly. This analysis-synthesis method is quite suitable for the method proposed in this paper.

The whole procedure for speech synthesis can be summarized as follows:

1. Training

(a) By using mel-cepstral analysis method, obtain mel-cepstral coefficients of speech database.
(b) Train phoneme HMMs.

2. Synthesis

(a) Concatenate phoneme HMMs according to the phoneme sequence translated from the text to be synthesized.
(b) Obtain a sequence of mel-cepstral coefficients vector by applying the proposed algorithm to the concatenated HMM.
(c) Synthesize speech by the MLSA filter.

HMMs can be context dependent, and the units of HMMs can be other than phonemes. Since the spectrum corresponding to $\boldsymbol{\mu}_{q,i}$ is the average of different speech spectra, its formant structure may become vague. A possible way to avoid this situation is to find

the best training vector which maximizes the output probability of the sub-state (q, i) and substitute it for $\boldsymbol{\mu}_{q,i}$.

5. EXAMPLE

A simple experiment of speech synthesis was carried out using the ATR Japanese data base. Data from one speaker (speaker MHT) was used. We used 25 different phoneme models plus an additional silence model `sil`, i.e., 26 models in all. The type of HMM used was a continuous Gaussian 3-mixture model. The diagonal covariances were used. All models were 3-state left to right models with no skips. The heuristic duration densities were calculated after the training. The feature vector comprised of 16 mel-cepstral coefficients including the 0-th coefficient, and their delta and delta-delta coefficients. Mel-cepstral coefficients were obtained by the mel-cepstral analysis. The signal was windowed by a 25.6ms Blackman window with a 5ms shift.

Fig. 1 shows the spectra calculated from the mel-cepstral coefficients generated by the HMM, which is composed by concatenation of phoneme models; `sil`, `n`, `a`, `y`, `a`, `m`, `u`, `sil`. Without the dynamic features, the parameter sequence which maximizes $P[\mathbf{Q}, \mathbf{O} \mid \lambda]$ becomes a sequence of the mean vectors (see Fig. 1(a)). On the other hand, Fig. 1(b) and Fig. 1(c) show that appropriate parameter sequences are generated by using the static and dynamic features. Looking at Fig. 1(b) and Fig. 1(c) closely, we can see that incorporation of delta-delta parameter improves smoothness of generated speech spectra.

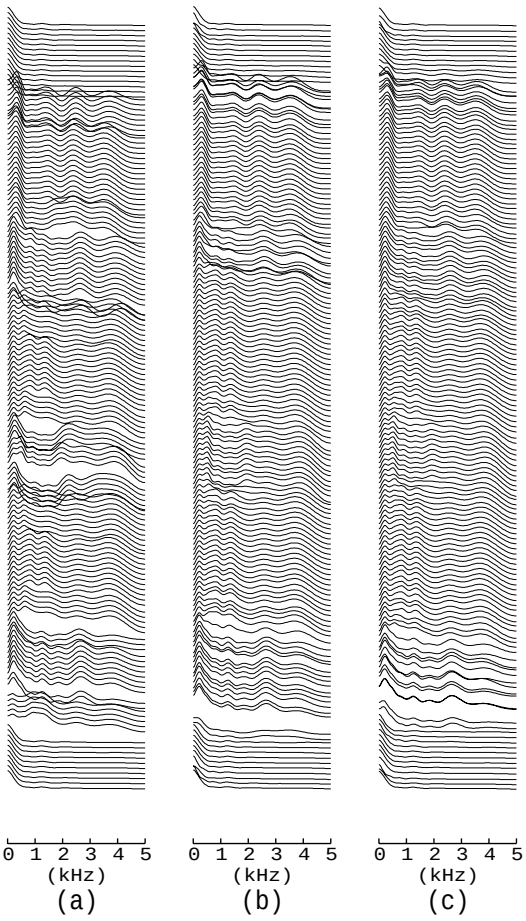

Figure 1: An example of parameter generation from an HMM composed by concatenation of phoneme models; `sil`, `n`, `a`, `y`, `a`, `m`, `u`, `sil`, (a) without dynamic features, (b) with delta, and (c) with delta and delta-delta

From informal listening tests of synthesized speech, it has been observed that without dynamic features discontinuity is perceptible whereas with dynamic features the synthesized speech is quite smooth. The deference between the speech quality with delta and that with delta and delta-delta is relatively small.

6. CONCLUSION

We have proposed an algorithm for speech parameter generation from continuous mixture HMMs using delta and delta-delta parameters. It has been shown that incorporating the dynamic features is essential to generate speech parameters from HMMs. It is expected that the algorithm is useful for speech synthesis by rule, speech enhancement, voice conversion, quantization of speech parameters, etc. Our future work will be further reduction of the computational complexity, and implementation of a speech synthesis system, which can synthesize speech with various voice qualities and emotions.

7. REFERENCES

[1] A. Ljolje and F. Fallside, "Synthesis of natural sounding pitch contous in isolated utterances using hidden Markov models," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp.1074–1080, 1986.

[2] M. Giustiniani, P. Pierucci, "Phonetic ergodic HMM for speech synthesis," in *Proc. EUROSPEECH-91*, 1991, pp.349–352.

[3] T. Fukada, Y. Komori, T. Aso and Y. Ohora, "A study of pitch pattern generation using HMM-based statistical information," in *Proc. ICSLP-94*, 1994, pp.723–726.

[4] R. E. Donovan and P. C. Woodland, "Automatic speech synthesiser parameter estimation using HMMs," in *Proc. ICASSP-95*, 1995, pp.640–643.

[5] E. P. Farges and M. A. Clements, "An analysis-synthesis hidden Markov model of speech," in *Proc. ICASSP-88*, 1988, pp.323–326.

[6] K. Tokuda, T. Kobayashi and S. Imai, "Speech parameter generation from HMM using dynamic features," in *Proc. ICASSP-95*, 1995, pp.660–663.

[7] S. Haykin, *Adaptive Filter Theory*, Englewood Cliffs, N.J.: Prentice-Hall, 1991.

[8] T. Fukada, K. Tokuda, T. Kobayashi and S. Imai, "An adaptive algorithm for mel-cepstral analysis of speech," in *Proc. ICASSP-92*, 1992, pp.I-137–I-140.

[9] S. Imai, "Cepstral analysis synthesis on the mel frequency scale," in *Proc. ICASSP-83*, 1983, pp.93–96.