Speaker Adaptation for Speech Synthesis Based on Deep Neural Networks Using Hidden Semi-Markov Model Structures

Kento Nakao*, Kei Hashimoto*, Keiichiro Oura*, Yoshihiko Nankaku*, and Keiichi Tokuda*
* Department of Computer Science, Nagoya Institute of Technology, Japan
E-mail: {kento0316, bonanza, uratec, nankaku, tokuda}@sp.nitech.ac.jp Tel: +81-52-735-5479

***Abstract*—This paper proposes a speaker adaptation technique for speech synthesis-based deep neural networks (DNNs) using hidden semi-Markov model (HSMM) structures. Speaker adaptation techniques for DNN-based speech synthesis are based on fixed time-alignments estimated by external aligners. Therefore, the acoustic features and temporal structures of speech are separately adapted in speaker adaptation. In this work, a special type of mixture density network (MDN) called MDN-HSMM, which outputs the parameters of HSMMs, is applied. The proposed method can model not only acoustic features but also durations in a unified framework and perform speaker adaptation that considers temporal structures. Experimental results show that the proposed method improves the naturalness and speaker similarity of the synthesized speech compared to the speaker adaptation based on DNNs.**

I. Introduction

In the last decade, statistical parametric speech synthesis (SPSS) has grown in popularity [1]. Hidden Markov model (HMM)-based speech synthesis [2] is one of the most popular approaches to SPSS, in which spectrum, fundamental frequency (F_0), and duration parameters are modeled in the unified framework [3] of a hidden semi-Markov model (HSMM), a special type of HMM that has explicit state duration probability distributions [4]–[7].

Recently, SPSS using deep neural networks (DNNs) has shown the potential to produce natural-sounding synthesized speech. A number of studies have demonstrated that DNN-based speech synthesis can achieve significantly better performance than conventional HMM-based speech synthesis [8]–[12]. Mixture density networks (MDNs) have been also applied to speech synthesis [13]. While DNNs predict only the mean parameters of output probability distributions, MDNs predict both mean and covariance parameters. The use of the mixture density output layer improves the prediction accuracy of acoustic features and the naturalness of the synthesized speech.

Apart from improving the naturalness of synthesized speech, speech synthesis systems are also expected to be able to generate an arbitrary speaker's voice with only a small amount of data by using a technique called speaker adaptation. Recently, a number of speaker adaptation methods for DNN-based speech synthesis have been proposed [14]. Generally, there are three ways to control the speaker identity in a DNN-based acoustic model. One way is to control the speaker identity at the input layer, such as adding speaker information as auxiliary input features [15]–[17]. The second way is to perform speaker adaptation with specially designed hidden layers, such as learning hidden unit contribution (LHUC) [18]. The third way is to adapt output features, such as speaker-dependent regression or feature space transformation [19].

In conventional DNN-based speech synthesis approaches, spectrum and F_0 parameters are modeled by neural networks while the temporal structures of speech are modeled by external duration models; in other words, acoustic features and duration are independently modeled from time-aligned data. Thus, the temporal structures of speech are not considered during the training of DNN-based acoustic models. To address these limitations, a speech synthesis approach based on a novel neural network called MDN-HSMM has been proposed [20]. In this approach, spectrum, F_0, and duration are simultaneously modeled in a unified framework. MDN-HSMM, which has a special type of MDN structure, generates the parameters of an HSMM that can represent utterance-level probability density functions conditioned on the corresponding input feature sequence. Therefore, MDN-HSMM can model not only spectrum and F_0 parameters but also durations.

This paper proposes a speaker adaptation technique for MDN-HSMM-based speech synthesis. Speaker adaptation techniques for DNN-based speech synthesis are based on fixed time alignments, which means that the acoustic features and temporal structures of speech are separately adapted. In the proposed method, speaker adaptation considering temporal structures can be performed because MDN-HSMMs can model acoustic features and temporal structures in a unified framework. Although there are many speaker adaptation approaches, a technique that controls the speaker identity at the input layer by adding speaker information as a speaker code is applied in this paper.

Section 2 of this paper gives an overview of speech synthesis based on DNNs. Section 3 describes speech synthesis based on MDN-HSMMs and Section 4 describes speaker adaptation for speech synthesis based on MDN-HSMMs. Experimental results are presented in Section 5. Concluding remarks are given in the final section.

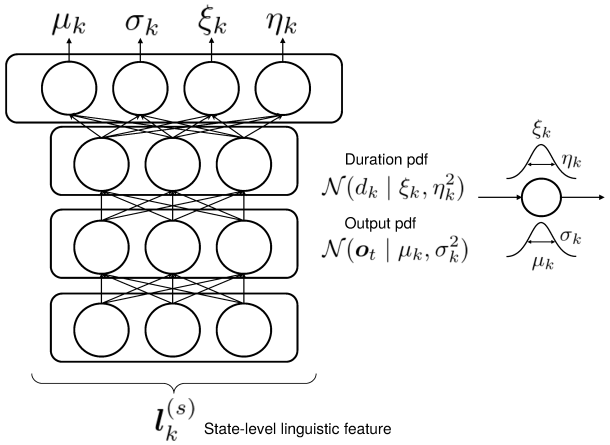

Fig. 1. Model structure of MDN-HSMM.

II. DNN-BASED SPEECH SYNTHESIS

The standard DNN-based speech synthesis consists of training and synthesis parts. In the training part, first, acoustic feature sequences and linguistic feature sequences are extracted. A frame-level acoustic feature sequence is extracted from speech waveform

$$\boldsymbol{o}^{(f)} = (\boldsymbol{o}_1^{(f)}, \boldsymbol{o}_2^{(f)}, \ldots, \boldsymbol{o}_T^{(f)}), \tag{1}$$

where T is the number of frames extracted from the speech waveform. In addition, a phoneme-level linguistic feature sequence is extracted from a text

$$\boldsymbol{l}^{(p)} = (\boldsymbol{l}_1^{(p)}, \boldsymbol{l}_2^{(p)}, \ldots, \boldsymbol{l}_I^{(p)}), \tag{2}$$

where I is the number of phonemes included in the text. Acoustic models in statistical parametric speech synthesis represent the relation between linguistic and acoustic feature sequences. However, it is difficult to represent the relation between phoneme-level linguistic feature sequences and frame-level acoustic feature sequences directly. To avoid this problem, the phoneme-level linguistic feature sequence $\boldsymbol{l}^{(p)}$ is converted into a frame-level linguistic feature sequence

$$\boldsymbol{l}^{(f)} = (\boldsymbol{l}_1^{(f)}, \boldsymbol{l}_2^{(f)}, \ldots, \boldsymbol{l}_T^{(f)}). \tag{3}$$

Then, the frame-level relation between linguistic and acoustic features is modeled by a DNN. The frame-level linguistic features are obtained according to the phoneme alignment estimated in advance. Therefore, temporal structures are not considered in the training of DNN.

III. MDN-HSMM-BASED SPEECH SYNTHESIS

A. Model structure

Speech synthesis based on HSMMs can simultaneously model acoustic feature sequences and duration of speech. In contrast, in the standard DNN-based speech synthesis, acoustic features are modeled by a DNN and duration information is modeled by an external duration model. To simultaneously model the acoustic features and temporal structures of speech based on DNNs in a unified framework, a speech synthesis approach based on a special type of MDN called MDN-HSMM, which outputs the parameters of an HSMM, has been proposed [20]. Figure 1 shows the model structure of the MDN-HSMM. It can model the conditional probability distributions of output feature sequences given input feature sequences by using the structure of an HSMM. The utterance-level likelihood used in the MDN-HSMM is defined as

$$\begin{aligned} &p(\boldsymbol{o}^{(f)} \mid \boldsymbol{l}^{(s)}, \boldsymbol{\lambda}^{(s)}) \\ &= \sum_{\boldsymbol{q}} p(\boldsymbol{o}^{(f)} \mid \boldsymbol{q}, \boldsymbol{l}^{(s)}, \boldsymbol{\lambda}^{(s)}) p(\boldsymbol{q} \mid \boldsymbol{l}^{(s)}, \boldsymbol{\lambda}^{(s)}) \\ &= \sum_{\boldsymbol{q}} \left\{ \prod_{t=1}^{T} p(\boldsymbol{o}_t^{(f)} \mid \boldsymbol{q}_t, \boldsymbol{l}^{(s)}, \boldsymbol{\lambda}^{(s)}) \prod_{k=1}^{K} p(d_k \mid k, \boldsymbol{l}^{(s)}, \boldsymbol{\lambda}^{(s)}) \right\} \end{aligned} \tag{4}$$

where $\boldsymbol{l}^{(s)}$ is a state-level linguistic feature sequence, d_k is a state duration, $\boldsymbol{q}$ is a state sequence, K is the number of states in the utterance, i.e., $K = I \cdot J$, and J is the number of states in each phoneme. The state duration is determined from the state sequence as

$$\begin{aligned} \boldsymbol{q} &= (q_1, q_2, \ldots, q_T) \\ &= (\underbrace{1, \ldots, 1}_{d_1}, \underbrace{2, \ldots, 2}_{d_2}, \ldots\ldots, \underbrace{K, \ldots, K}_{d_K}). \end{aligned} \tag{5}$$

The linguistic feature sequence

$$\boldsymbol{l}^{(s)} = (\boldsymbol{l}_1^{(s)}, \boldsymbol{l}_2^{(s)}, \ldots, \boldsymbol{l}_K^{(s)}) \tag{6}$$

is fed into the neural network in a state manner, and the neural network outputs state Gaussian for acoustic features $\mathcal{N}(\boldsymbol{o}_t \mid \boldsymbol{\mu}_{q_t}, \boldsymbol{\sigma}_{q_t}^2)$ and Gaussian for state duration $\mathcal{N}(d_k \mid \xi_k, \eta_k^2)$, i.e.,

$$p(\boldsymbol{o}_t^{(f)} \mid q_t, \boldsymbol{l}^{(s)}, \boldsymbol{\lambda}^{(s)}) = \mathcal{N}(\boldsymbol{o}_t^{(f)} \mid \boldsymbol{\mu}_{q_t}, \boldsymbol{\sigma}_{q_t}^2) \tag{7}$$

$$p(d_k \mid k, \boldsymbol{l}^{(s)}, \boldsymbol{\lambda}^{(s)}) = \mathcal{N}(d_k \mid \xi_k, \eta_k^2). \tag{8}$$

The Gaussian parameters can be derived from the MDN as

$$\boldsymbol{\mu}_k = z^{(\mu)}(\boldsymbol{l}_k^{(s)}) \tag{9}$$

$$\boldsymbol{\sigma}_k = \exp(z^{(\sigma)}(\boldsymbol{l}_k^{(s)})) \tag{10}$$

$$\xi_k = z^{(\xi)}(\boldsymbol{l}_k^{(s)}) \tag{11}$$

$$\eta_k = \exp(z^{(\eta)}(\boldsymbol{l}_k^{(s)})) \tag{12}$$

where $z^{(\cdot)}(\boldsymbol{l}_k^{(s)})$ is the activation of the output layer of the MDN corresponding to each parameter when a state-level linguistic feature sequence is fed into the MDN. The conversion from a phoneme-level feature sequence to a state-level feature sequence is straightforward because each phoneme always consists of a fixed number of states.

B. Training part

The training of MDN-HSMM aims to estimate the parameters of MDN-HSMM that maximize the likelihood defined in (4). However, it is difficult to maximize the likelihood of

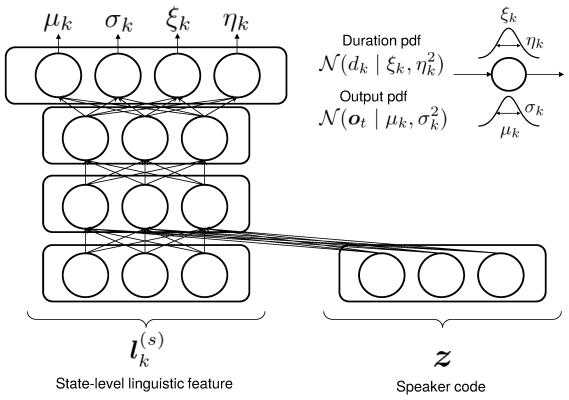

Fig. 2. MDN-HSMM-based speech synthesis system using speaker code.

MDN-HSMM directly. Therefore, a $\mathcal{Q}$ function is used for the training.

$$\mathcal{Q}(\bar{\boldsymbol{\lambda}}^{(s)}, \boldsymbol{\lambda}^{(s)}) = \sum_{\boldsymbol{q}} p(\boldsymbol{q} \mid \boldsymbol{o}^{(f)}, \boldsymbol{l}^{(s)}, \bar{\boldsymbol{\lambda}}^{(s)}) \times \log p(\boldsymbol{o}^{(f)}, \boldsymbol{q} \mid \boldsymbol{l}^{(s)}, \boldsymbol{\lambda}^{(s)}) \quad (13)$$

Posterior probabilities $p(\boldsymbol{q} \mid \boldsymbol{o}^{(f)}, \boldsymbol{l}^{(s)}, \bar{\boldsymbol{\lambda}}^{(s)})$ are efficiently calculated by the generalized forward-backward algorithm. By using the negative $\mathcal{Q}$ function as the error function and back-propagating the derivatives of the negative $\mathcal{Q}$ function through the network, the neural network weights can be updated to maximize the log likelihood.

C. Synthesis part

First, a given text to be synthesized is converted into a state-level linguistic feature sequence. Second, the state-level linguistic feature sequence is fed into the neural network and the parameters of the HSMMs are predicted. Third, the state sequence $\hat{\boldsymbol{q}}$ is determined by the estimated duration distributions of HSMMs as

$$\hat{\boldsymbol{q}} = \arg\max_{\boldsymbol{q}} p(\boldsymbol{q} \mid \boldsymbol{l}^{(s)}, \boldsymbol{\lambda}^{(s)}) = \arg\max_{\boldsymbol{q}} \prod_{k=1}^{K} p(d_k \mid k, \boldsymbol{l}_k^{(s)}, \boldsymbol{\lambda}^{(s)}). \quad (14)$$

Finally, the acoustic feature sequence $\hat{\boldsymbol{o}}^{(f)}$ is obtained by maximizing the output probability given the estimated state sequence $\hat{\boldsymbol{q}}$ as

$$\hat{\boldsymbol{o}}^{(f)} = \arg\max_{\boldsymbol{o}^{(f)}} p(\boldsymbol{o}^{(f)} \mid \hat{\boldsymbol{q}}, \boldsymbol{l}^{(s)}, \boldsymbol{\lambda}^{(s)}). \quad (15)$$

IV. Speaker adaptation for speech synthesis based on MDN-HSMM

Recently, a number of speaker adaptation methods for DNN-based speech synthesis have been proposed [15]. In this paper, speaker adaptation based on speaker codes is applied to speech synthesis based on MDN-HSMMs. A speaker code is represented as an N-dimensional vector $\boldsymbol{z} = [z_1, z_2, \ldots, z_N]^\top$. While DNN-based speech synthesis models acoustic features on the basis of pre-calculated duration information, MDN-HSMMs can simultaneously model acoustic features and temporal structures in a unified framework. Thus, in the proposed method, speaker adaptation taking account of acoustic features and temporal structures can be performed. The proposed method consists of two parts: multi-speaker modeling and speaker adaptation.

A. Multi-speaker modeling based on MDN-HSMM

1) Training: In the training part of the proposed method, training data including multiple speakers' speech data is used. Input features for the proposed method consist of state-level linguistic features and a speaker code $\boldsymbol{z}$. The speaker code $\boldsymbol{z}$ for speaker a included in the training data is represented as a one-hot vector

$$z_n = \begin{cases} 1 = (n = a) \\ 0 = (n \neq a) \end{cases}, \quad (16)$$

where the number of dimensions N for the speaker code is equal to the number of speakers in the training data. The MDN-HSMM-based speech synthesis system using speaker code is presented in Fig. 2. In the proposed method, the speaker code is directly fed to the first hidden layer through a set of new connection weights. The likelihood function for MDN-HSMMs with speaker codes is defined as

$$p(\boldsymbol{o}^{(f)} \mid \boldsymbol{l}^{(s)}, \boldsymbol{z}, \boldsymbol{\lambda}^{(s)}) = \sum_{\boldsymbol{q}} \left\{ \prod_{t=1}^{T} p(\boldsymbol{o}_t^{(f)} \mid \boldsymbol{q}_t, \boldsymbol{l}^{(s)}, \boldsymbol{z}, \boldsymbol{\lambda}^{(s)}) \prod_{k=1}^{K} p(d_k \mid k, \boldsymbol{l}^{(s)}, \boldsymbol{z}, \boldsymbol{\lambda}^{(s)}) \right\}. \quad (17)$$

The parameters of MDN-HSMM are estimated with consideration of the speaker characteristics by maximizing the $\mathcal{Q}$ function.

$$\hat{\boldsymbol{\lambda}}^{(s)} = \arg\max_{\boldsymbol{\lambda}^{(s)}} \mathcal{Q}(\bar{\boldsymbol{\lambda}}^{(s)}, \boldsymbol{\lambda}^{(s)}) = \arg\max_{\boldsymbol{\lambda}^{(s)}} \sum_{\boldsymbol{q}} p(\boldsymbol{q} \mid \boldsymbol{o}^{(f)}, \boldsymbol{l}^{(s)}, \boldsymbol{z}, \bar{\boldsymbol{\lambda}}^{(s)}) \times \log p(\boldsymbol{o}^{(f)}, \boldsymbol{q} \mid \boldsymbol{l}^{(s)}, \boldsymbol{z}, \boldsymbol{\lambda}^{(s)}) \quad (18)$$

MDN-HSMM is trained by the BP algorithm and the generalized forward-backward algorithm.

2) Synthesis: A state-level linguistic feature sequence $\boldsymbol{l}^{(s)}$ to be synthesized and the speaker code representing the target speaker $\boldsymbol{z}$ are fed into the trained MDN-HSMM. Then, the state sequence $\hat{\boldsymbol{q}}$ for the target speaker is determined by the duration distributions generated by the MDN-HSMM with the speaker code.

$$\hat{\boldsymbol{q}} = \arg\max_{\boldsymbol{q}} p(\boldsymbol{q} \mid \boldsymbol{l}^{(s)}, \boldsymbol{z}, \boldsymbol{\lambda}^{(s)}) = \arg\max_{\boldsymbol{q}} \prod_{k=1}^{K} p(d_k \mid k, \boldsymbol{l}_k^{(s)}, \boldsymbol{z}, \boldsymbol{\lambda}^{(s)}) \quad (19)$$

If the target speaker is a speaker included in the training data, the speaker code representing the target speaker in the training part is used. The acoustic feature sequence $\hat{\boldsymbol{o}}^{(f)}$ for the target speaker is obtained by maximizing the output probability given the state sequence and the speaker code representing the target speaker

$$\hat{\boldsymbol{o}}^{(f)} = \arg\max_{\boldsymbol{o}^{(f)}} p(\boldsymbol{o}^{(f)} \mid \hat{\boldsymbol{q}}, \boldsymbol{l}^{(s)}, \boldsymbol{z}, \boldsymbol{\lambda}^{(s)}). \tag{20}$$

B. Speaker adaptation based on speaker codes

In the adaptation part, a new speaker code for a new target speaker is estimated from adaptation data for the target speaker on the basis of the $\mathcal{Q}$ function

$$\begin{aligned}\hat{\boldsymbol{z}} &= \arg\max_{\boldsymbol{z}} \mathcal{Q}(\bar{\boldsymbol{z}}, \boldsymbol{z}) \\ &= \arg\max_{\boldsymbol{z}} \sum_{\boldsymbol{q}} p(\boldsymbol{q} \mid \tilde{\boldsymbol{o}}^{(f)}, \tilde{\boldsymbol{l}}^{(s)}, \bar{\boldsymbol{z}}, \boldsymbol{\lambda}^{(s)}) \\ &\quad \times \log p(\tilde{\boldsymbol{o}}^{(f)}, \boldsymbol{q} \mid \tilde{\boldsymbol{l}}^{(s)}, \boldsymbol{z}, \boldsymbol{\lambda}^{(s)})\end{aligned} \tag{21}$$

where $\boldsymbol{o}, \boldsymbol{l}$ are the acoustic and linguistic feature sequences extracted from the adaptation data. The speaker code for the target speaker can be estimated on the basis of the generalized forward backward algorithm and the BP algorithm. During this phase, the weight parameters of MDN-HSMM are kept unchanged, and only the speaker code is estimated from the adaptation data. Since the parameters of the HSMMs, which are the output of MDN-HSMM, change according to the input, it is possible to obtain the parameters of the HSMM expressing the characteristics of the target speaker by estimating an appropriate speaker code.

In the synthesis part, by feeding linguistic features and the estimated speaker code $\hat{\boldsymbol{z}}$ into the trained MDN-HSMM, the output probability distributions and the duration probability distributions, which construct HSMMs for the target speaker, are generated. The state sequence and the acoustic features are obtained by using the generated distributions as

$$\begin{aligned}\hat{\boldsymbol{q}} &= \arg\max_{\boldsymbol{q}} p(\boldsymbol{q} \mid \boldsymbol{l}^{(s)}, \hat{\boldsymbol{z}}, \boldsymbol{\lambda}^{(s)}) \\ &= \arg\max_{\boldsymbol{q}} \prod_{k=1}^{K} p(d_k \mid k, \boldsymbol{l}_k^{(s)}, \hat{\boldsymbol{z}}, \boldsymbol{\lambda}^{(s)})\end{aligned} \tag{22}$$

$$\hat{\boldsymbol{o}}^{(f)} = \arg\max_{\boldsymbol{o}^{(f)}} p(\boldsymbol{o}^{(f)} \mid \hat{\boldsymbol{q}}, \boldsymbol{l}^{(s)}, \hat{\boldsymbol{z}}, \boldsymbol{\lambda}^{(s)}). \tag{23}$$

V. EXPERIMENTS

In order to determine the effectiveness of the proposed method, objective and subjective evaluations were conducted. In these evaluations, two speech synthesis systems were compared: the DNN-based system and the MDN-HSMM-based system.

A. Experimental setup

A Japanese speech database constructed by our research group was used in the experiments. The database contains a set of 503 phonetically balanced uttered sentences. The set is the same as the B-set of the ATR phonetically balanced Japanese speech database [21]. In the experiments, speech data uttered by 61 speakers was used. One male speaker included in the speech data was the target speaker. As the training data, three datasets were prepared: 1000 utterances from 20 speakers, 2000 utterances from 40 speakers, and 3000 utterances from 60 speakers. The target speaker was not included in the training data, and each speaker has 50 utterances for training. The adaptation data was 25 utterances uttered by the target speaker and the test data was 53 utterances that were not included in the training or adaptation data.

Speech signals were sampled at 48 kHz. Acoustic feature vectors were extracted with a 5-ms shift and consisted of 0-th through 49-th mel-cepstral coefficients and a log F_0 value, which were normalized to have zero-mean unit-variance, dynamic features (delta and delta-delta), and a voiced/unvoiced binary value. Mel-cepstral coefficients were extracted from the smoothed spectrum analyzed by STRAIGHT [22].

The MDN-HSMM and DNN used in the experiments had three hidden layers with 1024 units per layer. The sigmoid activation function was used in the hidden layers. The linear activation function was used in the output layer of the DNN, and the activation functions defined in (9)–(12) were used in the output layer of the MDN-HSMM. The input feature for DNN was a 411-dimensional feature vector consisting of 408 linguistic features including binary features and numerical features and three duration features including duration of the current phoneme and the relative position of the current frame in the phoneme. In the MDN-HSMM-based system, a phoneme was represented by an HSMM with the five-state, left-to-right, no-skip structure. Therefore, 408 linguistic features and five binary features representing the state index in the phoneme were used as the input feature. The input features were normalized to be within 0.0-1.0 based on their minimum and maximum values in the training data. The duration information for the test data was derived from forced-alignment to natural speech with HSMMs, which were separately trained with the same training data, for DNN. In MDN-HSMM, the durations for each state were predicted by the duration distributions generated by the MDN-HSMM. During training and adaptation based on DNN, the minibatch size was set to 128. For the MDN-HSMM, one utterance was used as a minibatch in the training.

B. Objective evaluation

Objective evaluation to analyze the performance of each individual adaptation was conducted. To objectively evaluate the performance of the systems, the mel-cepstral distortion (MCD) was used as an objective measure. The MCD was calculated by

$$\mathrm{MCD} = \frac{10}{\log 10} \sqrt{2 \sum_{d=1}^{D} \left(\boldsymbol{c}_d^{(tar)} - \boldsymbol{c}_d^{(syn)} \right)^2}, \tag{24}$$

where $\boldsymbol{c}_d^{(tar)}$ and $\boldsymbol{c}_d^{(syn)}$ are the d^{th} coefficients of the target and synthesized mel-cepstrum and D is the order of mel-cepstrum. Three datasets–1000 utterances from 20 speakers, 2000 utterances from 40 speakers, and 3000 utterances from

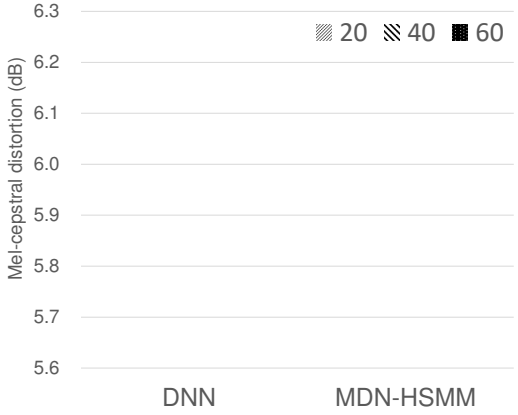

Fig. 3. Schematic diagram of speech production. (20, 40, 60 speakers were used for training.)

60 speakers–were used for training. Since the number of dimensions of the speaker code depends on the number of speakers in the training data, 20-, 40-, and 60-dimensional speaker codes were used for each training dataset, respectively.

The experimental results are shown in Fig. 3. The MCDs of MDN-HSMM-based systems were lower than the ones of DNN-based systems under all conditions. This is because the MDN-HSMM-based method has an advantage over the DNN-based method in that the speaker code for the target speaker can be estimated taking account of not only acoustic features but also temporal structures. Also, as the amount of training data was expanded from 1000 to 3000 utterances, both the DNN-based and MDN-HSMM-based methods reduced MCDs. These results indicate that the speaker adaptation based on speaker codes improved the MCDs by increasing the number of speakers included in the training data, and the proposed method can output more appropriate speaker codes than the DNN-based method under all conditions.

C. Subjective evaluation

Next, subjective evaluations by listening tests to assess the naturalness and speaker similarity of the synthesized speech were conducted. In the listening test for naturalness, two speech samples synthesized by the DNN-based and MDN-HSMM-based systems were played in randomized order, and participants were asked which of the two samples sounded more natural. In the listening test for speaker similarity, a reference speech of the target speaker was first played and then two synthesized speech samples were played in randomized order. Participants were asked which of the two samples sounded more similar to the reference speech. Speech samples used in these experiments were synthesized by the systems trained with 3000 utterances from 60 speakers. Ten Japanese listeners participated in the test. Each listener rated 20 sets that were randomly selected from the testing utterances.

TABLE I
PREFERENCE SCORES BETWEEN DNN AND MDN-HSMM ADAPTATIONS.

	DNN	MDN-HSMM	Neutral
Speaker similarity	14.5%	82.0%	3.5%
Naturalness	11.0%	80.0%	9.0%

Subjective evaluation results are presented in Table I. The proposed method achieved significantly better performance than the conventional DNN-based method in terms of both naturalness and speaker similarity. Although the DNN-based method estimates speaker codes with fixed time-alignment information obtained by the external aligner, such as HSMMs, the proposed method can estimate appropriate speaker codes because the MDN-HSMM can model not only spectrum and F_0 but also the duration of speech simultaneously in a unified framework. Thus, the synthesized speech based on the proposed approach came closer to the target speech in terms of durations of speech than the DNN-based approach.

VI. CONCLUSIONS

In this paper, speaker adaptation for speech synthesis based on a neural network that outputs the parameters of probability distributions constructing HSMMs, called MDN-HSMM, was proposed. In the proposed method, speaker adaptation based on speaker codes was applied to the MDN-HSMM-based speech synthesis. The MDN-HSMM-based speaker adaptation method can estimate appropriate speaker codes for the target speaker by taking account of the temporal structures of speech in a unified framework based on HSMMs. This is in contrast to the DNN-based method, which cannot consider temporal structures because it uses the fixed time-alignment information obtained by external duration models. Experimental results show that the proposed approach improves the naturalness and speaker similarity of synthesized speech compared to the conventional DNN-based approach.

Further analysis will involve investigation of the effects of training data, e.g., the amount of training data and the number of speakers included in the training data, and how to improve the quality of speaker adaptation for MDN-HSMM-based speech synthesis using other speaker adaptation methods.

ACKNOWLEDGMENTS

This research and development work was partly supported by the MIC/SCOPE #162106106

REFERENCES

[1] H. Zen, K. Tokuda, and A.W. Black,"Statistical parametric speech synthesis,"*Speech Communication*, vol. 51, no. 11, pp. 1039–1064, 2009.
[2] K. Tokuda, Y. Nankaku, T. Toda, H. Zen, J. Yamagishi, and K. Oura, "Speech synthesis based on hidden Markov models," *Proceedings of IEEE*, vol. 101, no. 5, pp. 1234–1252, 2013.
[3] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "A hidden semi-Markov model-based speech synthesis system," *IEICE Trans. Inf. Syst.*, vol. E90-D, no. 5, pp. 825–834, 2007.
[4] J. Ferguson, "Variable duration models for speech," *Proceedings of the Symposium on the Application Hidden Markov Models to Text and Speech*, 1980, pp. 143–179.
[5] M. Russell and R. Moore, "Explicit modeling of state occupancy in hidden Markov models for automatic speech recognition," *Proceedings of ICASSP*, 1985, pp. 5–8.

[6] S. Levinson, "Continuously variable duration hidden Markov models for automatic speech recognition," *Comput. Speech Lang.*, vol. 1, pp. 29–45, 1986.
[7] C. Mitchell, M. Harper, and L. Jamieson, "On the complexity of explicit duration HMMs," *IEEE Trans. Speech Audio Process.*, vol. 3, no. 3, pp. 213–217, 1995.
[8] H. Zen, A. Senior, and M.Schuster, "Statistical parametric speech synthesis using deep neural networks," *Proceedings of ICASSP 2013*, pp. 7962–7966, 2013.
[9] Z.-H. Ling, S. Kang, H. Zen, A. Senior, M. Schuster, X.-J. Qian, H. Meng, and L. Deng, "Deep learning for acoustic modeling in parametric speech generation: A systematic review of existing techniques and feature trends," *IEEE Signal Processing Magazine*, vol. 32, no. 5, pp. 35–52, 2015.
[10] Y. Qian, Y. Fan, H. Wenping, and F.K. Soong, "On the training aspects of deep neural network (DNN) for parametric TTS synthesis," *Proceedings of ICASSP 2014*, pp. 3857–3861, 2014.
[11] K. Hashimoto, K. Oura, Y. Nankaku, and K. Tokuda, "The effect of neural networks in statistical parametric speech synthesis," *Proceedings of ICASSP 2015*, pp. 4455–4459, 2015.
[12] S. Takaki, S.-J Kim, J. Yamagishi, and J.-J. Kim, "Multiple feed-forward deep neural networks for statistical para-metric speech synthesis," *Proceedings of Interspeech 2015*, pp. 2242–2246, 2015.
[13] H. Zen, A. Senior, "Deep Mixture Density Networks for Acoustic Modeling in Statistical Parametric Speech Synthesis," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, IEEE (2014), pp. 3872–3876.
[14] Z. Wu, P. Swietojanski, C. Veaux, S.Renals, and S. King, "A study of speaker adaptation for DNN-based speech synthesis," *Proceedings of Interspeech 2015*, pp. 879–883, 2015.
[15] O. Abdel-Hamid and H. Jiang, "Fast speaker adaptation of hybrid NN/HMM model for speech recognition based on discriminative learning of speaker code," *Proceedings of 2013 IEEE Inter-national Conference on Acousitcs, Speech and Signal Processing (ICASSP)*, pp. 7942–7946, 2013.
[16] S. Xue, O. Abdel-Hamid, H. Jiang, L. Dai, and Q. Liu, "Fast adaptation of deep neural network based on discriminant codes for speech recognition," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 12, pp. 1713–1725, 2014.
[17] N. Hojo, Y. Ijima, and H. Mizuno, "An investigation of DNN-based speech synthesis using speaker codes," *Proceedings of Interspeech 2016*, pp. 2278–2282, 2016.
[18] P. Swietojanski and S. Renals, "Learning hidden unit contributions for unsupervised speaker adaptation of neural network acoustic models," *Spoken Language Tehnology Work-shop (SLT), 2014 IEEE*, pp. 171–176, 2014.
[19] Y. Fan, Y. Qian, F.K. Soong, and L. He, "Multi-speaker modeling and speaker adaptation for DNN-based tts synthesis," *Proceedings of 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) IEEE*, pp. 4475–4479, 2015.
[20] K. Tokuda, K. Hashimoto, K. Oura, and Y. Nankaku, "Temporal modeling in neural network based statistical parametric speech synthesis," *9th ISCA Speech Synthesis Workshop*, pp. 113–118, Sunnyvale, USA, September, 2016.
[21] A. Kurematsu, K. Takeda, Y. Sagisaka, S. Katagiri, H. Kuwabara, and K. Shikan, "ATR Japanese speech database as a tool of speech recognition and synthesis," *Speech Communication*, pp. 357–363, 1990.
[22] H. Kawahara, I. Masuda-Katsuse, and A. Cheveigne, "Re- structuring speech representations using a pitch-adaptive time- frequency smoothing and an instantaneous-frequency-based f0 extraction: Possible role of a repetitive structure in sounds" *Speech Communication*, pp. 187–207, 1999.