PAPER

Generative Moment Matching Network-Based Neural Double-Tracking for Synthesized and Natural Singing Voices

Hiroki TAMARU[†a)], ***Nonmember***, **Yuki SAITO**[†b)], ***Student Member***, **Shinnosuke TAKAMICHI**[†c)], **Tomoki KORIYAMA**[†d)], ***and*** **Hiroshi SARUWATARI**[†e)], ***Members***

SUMMARY This paper proposes a generative moment matching network (GMMN)-based post-filtering method for providing *inter-utterance pitch variation* to singing voices and discusses its application to our developed mixing method called *neural double-tracking* (NDT). When a human singer sings and records the same song twice, there is a difference between the two recordings. The difference, which is called inter-utterance variation, enriches the performer's musical expression and the audience's experience. For example, it makes every concert special because it never recurs in exactly the same manner. Inter-utterance variation enables a mixing method called double-tracking (DT). With DT, the same phrase is recorded twice, then the two recordings are mixed to give richness to singing voices. However, in synthesized singing voices, which are commonly used to create music, there is no inter-utterance variation because the synthesis process is deterministic. There is also no inter-utterance variation when only one voice is recorded. Although there is a signal processing-based method called artificial DT (ADT) to layer singing voices, the signal processing results in unnatural sound artifacts. To solve these problems, we propose a post-filtering method for randomly modulating synthesized or natural singing voices as if the singer sang again. The post-filter built with our method models the inter-utterance pitch variation of human singing voices using a conditional GMMN. Evaluation results indicate that 1) the proposed method provides perceptible and natural inter-utterance variation to synthesized singing voices and that 2) our NDT exhibits higher *double-trackedness* than ADT when applied to both synthesized and natural singing voices.

key words: *DNN-based singing-voice synthesis, generative moment matching network, inter-utterance pitch variation, artificial double-tracking, modulation spectrum*

1. Introduction

When a person sings the same song twice, the resulting singing voices are never the same. This difference is called *inter-utterance variation* [1]. Inter-utterance variation leads to rich musical experiences. For example, when a singer's singing is different from that in a compact disc recording, the audience feels that the singer is really singing in front of them and can be moved by the singer's performance. Inter-utterance variation also enables choosing a favorite from various recordings of the same song in music production. A mixing method called double-tracking (DT) [2], [3] uses inter-utterance variation (Fig. 1). DT is achieved by layering two or more recordings by one singer to obtain a perceptually rich sound.

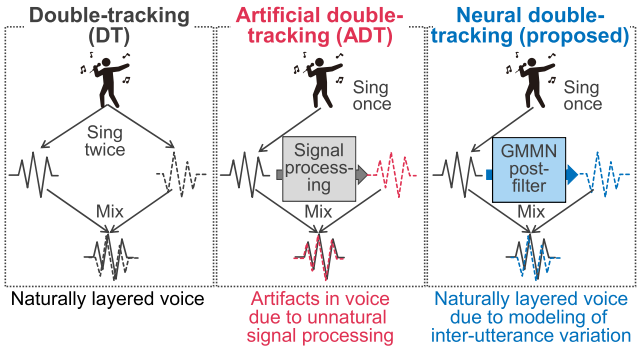


Fig. 1 Double-tracking (DT), artificial double-tracking (ADT), and our neural double-tracking (NDT). ADT and NDT are used for natural voices but can also be used for synthesized voices.

Singing-voice synthesis methods are currently used for creating music. One aim with such methods is the creation of expressive singing voices not depending on the creators' gender or singing ability. VOCALOID [4], which is a singing-voice synthesis software, is so popular that many creators upload original songs created using VOCALOID online and that there are even concerts performed by VOCALOID characters. Also, some industrial companies are aiming at developing the synthesis systems [5], [6]. There are a variety of singing-voice synthesis methods, e.g., unit selection synthesis (e.g., VOCALOID), those based on hidden Markov models (HMMs) [7], [8], and those based on deep neural networks (DNNs) [9], [10]. The performance of DNN-based ones is improving rapidly in recent years, outperforming HMM-based ones [9]. However, such DNN-based methods lack inter-utterance variation, as shown in Fig. 2. A single voice is synthesized from one musical score with such DNN-based methods because of the deterministic synthesis process. This results in the lack of the rich musical experiences mentioned above and makes DT impossible. The same problem also arises with human singers when there is only one successful recording but the creator wants to achieve double-trackedness. An alternative to DT is signal processing-based *artificial* or *automatic double-tracking* (ADT) [2], [11]. ADT, which requires only one recording, deterministically modulates one recording and mixes the original and modulated voices

Manuscript received August 23, 2019.
Manuscript revised November 14, 2019.
Manuscript publicized December 23, 2019.
[†]The authors are with The University of Tokyo, Tokyo, 113–8656 Japan.
a) E-mail: hiroki_tamaru@ipc.i.u-tokyo.ac.jp
b) E-mail: yuuki_saito@ipc.i.u-tokyo.ac.jp
c) E-mail: shinnosuke_takamichi@ipc.i.u-tokyo.ac.jp
d) E-mail: tomoki_koriyama@ipc.i.u-tokyo.ac.jp
e) E-mail: hiroshi_saruwatari@ipc.i.u-tokyo.ac.jp
DOI: 10.1587/transinf.2019EDP7228

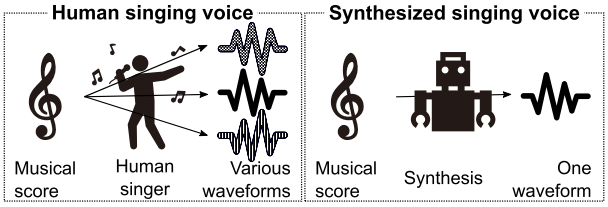


Fig. 2 Comparison of human and synthesized singing voices.

(Fig. 1). Since ADT does not require multiple recordings with inter-utterance variations, it can be easily applied to both human and synthesized singing voices. However, the signal processing of ADT results in unnatural sound artifacts [11].

To solve these problems, we propose a post-filtering method for providing inter-utterance pitch variations to synthesized and natural singing voices. We assume that such variations follow a certain complicated distribution; thus, we use deep generative models, which can model complicated distributions. Based on our previous work on spectrum generation in text-to-speech synthesis [1], we use a generative moment matching network (GMMN) [12], [13] as the deep generative model. This is because it is effective and easy to implement compared with other models. For instance, generative adversarial networks [14] involve a difficult minimax problem and variational auto-encoders [15] are subject to degradation of decoding quality due to over-regularization [16]. Our conditional GMMN-based post-filter is trained to represent the distribution of natural pitch contours given either synthesized or natural pitch contours and prior noise vectors. The trained post-filter randomly modulates any input pitch contour and outputs an alternative pitch contour as if the singer sang again. Instead of modeling the distribution of the pitch frame-wise, this post-filter models the distribution of the modulation spectrum (MS) [17] of the pitch contours so that a long-term pitch structure can be captured. We apply the proposed method to a new type of ADT we developed, called *neural double-tracking* (NDT). With NDT, the secondary singing voice is generated by modulating the input synthesized or natural voice using our method. Since our method provides natural inter-utterance variation, our NDT achieves naturally layered singing voices.

The experimental evaluation demonstrated that our method can provide perceptible inter-utterance pitch variations to synthesized singing voices while preserving naturalness and that the NDT exhibits higher perceptual double-trackedness than ADT in both synthesized and natural voices.

The rest of this paper is organized as follows. In Sect. 2, we describe two related methods, i.e., DNN-based singing-voice synthesis and ADT. In Sect. 3, we explain GMMN. In Sect. 4, we describe the proposed method. In Sects. 5 and 6, we present the experimental results for synthesized and natural voices, respectively. In Sect. 7, we conclude the paper. Note that this paper is partially based on an international conference paper [18] written by the authors, in which we discussed our method and NDT for synthesized singing voices. The contribution of this paper is applying our method to NDT for natural singing voices. We conduct experiments to evaluate the effectiveness of NDT for natural singing voices.

2. Related Methods

2.1 DNN-Based Singing-Voice Synthesis Method

With the DNN-based singing-voice synthesis method proposed by Nishimura et al. [9], the relationship between musical information and singing voices is modeled using DNNs. A musical score is converted into a sequence of vectors representing musical and linguistic contexts. A singing voice is converted into a sequence of speech parameters such as a fundamental frequency (F_0), spectral parameters, and an unvoiced/voiced label. In the training process, the mean squared error (MSE) between the predicted and target (natural) parameters is minimized as follows:

$$L_{\mathrm{MSE}}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = \|\boldsymbol{y} - \hat{\boldsymbol{y}}\|^2 \tag{1}$$

where $\boldsymbol{y}$ is the target and $\hat{\boldsymbol{y}}$ is the predicted speech-parameter sequences. We focus on 1-dimensional continuous F_0 [19] and define $\boldsymbol{y}$ as a scalar value sequence $[y(1), \cdots, y(t), \cdots, y(T)]^\top$, where $y(t)$ is the continuous log-scaled F_0 at frame t and $\top$ is the transpose. In the synthesis process, $\hat{\boldsymbol{y}} = [\hat{y}(1), \ldots, \hat{y}(t), \ldots, \hat{y}(T)]^\top$ is generated using the DNN given the input context vector sequence and $\hat{\boldsymbol{y}}$ is used to synthesize the output singing voice. The output voice is uniquely determined according to the input context since the synthesis process is deterministic.

2.2 ADT

DT is a mixing method used to provide richness to singing voices [2], [3] by recording and mixing the same musical phrase twice. The problem is that it is difficult for singers to avoid unnatural differences between the two recordings (e.g., note lengths). To solve this problem, ADT, which is an alternative method, was proposed. With ADT, instead of recording twice, the secondary voice is obtained by modulating the first voice based on signal processing [2], [11]. Originally, the process was achieved by taking a vocal signal from the sync head of a multi-track, recording it to another loop of tape which was speed varied with a slow oscillation, and recording it back onto the multi-track [2]. More recently, it has been replaced with computational signal processing; the most common method is the chorus effect [11]. This effect modulates the input pitch contour with a low-frequency oscillator (LFO). In other words, a simple function, such as a sine function, is added to the original pitch sequence. The modulated voice is then mixed with the original one. Typically, the modulated voice is given some volume reduction and temporal delay to enhance the doubled-voice

sense.

Although ADT can be applied to any singing voice including natural and synthesized voices, it involves mixing two signals with similar phases, resulting in comb-filtering and its subsequent unnatural changes of sound [11].

3. GMMN

A GMMN [12] is a neural network that enables random sampling from the target distribution. A GMMN takes a random noise vector as the input of the neural network where the noise behaves as the source of the randomness. In the training process, we minimize the maximum mean discrepancy (MMD) between the predicted and target distributions. MMD represents the discrepancy of statistical moments between the two distributions. In the generation process, a random value is sampled from the trained distribution.

A conditional GMMN [13] is an extension of a GMMN that enables random sampling from the conditional distribution given certain input variables. As the input of a conditional GMMN, the joint vector consisting of conditioning variables and random noise is used. The GMMN, which outputs a conditional distribution, is trained by minimizing conditional MMD (CMMD), which measures the distance between two conditional distributions. Here, we consider the CMMD between the distributions of the target vector included in training data and the output vector obtained by the neural network. Let $\boldsymbol{S}_{\text{in}} = [\boldsymbol{s}_{\text{in}}(1), \cdots, \boldsymbol{s}_{\text{in}}(T')]$ be the input conditioning vector set, $\boldsymbol{S}_{\text{tgt}} = [\boldsymbol{s}_{\text{tgt}}(1), \cdots, \boldsymbol{s}_{\text{tgt}}(T')]$ be the target vector set. T' is the total number of frames of training data. Let $\boldsymbol{S}_{\text{out}} = [\boldsymbol{s}_{\text{out}}(1), \cdots, \boldsymbol{s}_{\text{out}}(T')]$ be the output vector set and squared CMMD is defined as follows:

$$L(\boldsymbol{S}_{\text{in}}, \boldsymbol{S}_{\text{tgt}}, \boldsymbol{S}_{\text{out}}) = \frac{1}{T'^2} \|\boldsymbol{C}_{\boldsymbol{S}_{\text{out}}|\boldsymbol{S}_{\text{in}}} - \boldsymbol{C}_{\boldsymbol{S}_{\text{tgt}}|\boldsymbol{S}_{\text{in}}}\|^2 \tag{2}$$

where $\boldsymbol{C}_{\boldsymbol{S}_{\text{out}}|\boldsymbol{S}_{\text{in}}}$ and $\boldsymbol{C}_{\boldsymbol{S}_{\text{tgt}}|\boldsymbol{S}_{\text{in}}}$ are the covariance operators of $\boldsymbol{S}_{\text{out}}$ and $\boldsymbol{S}_{\text{tgt}}$ conditioned with $\boldsymbol{S}_{\text{in}}$, respectively. The estimates of $\boldsymbol{C}_{\boldsymbol{S}_{\text{out}}|\boldsymbol{S}_{\text{in}}}$ and $\boldsymbol{C}_{\boldsymbol{S}_{\text{tgt}}|\boldsymbol{S}_{\text{in}}}$ are given by the following equations [20]:

$$\hat{\boldsymbol{C}}_{\boldsymbol{S}_{\text{out}}|\boldsymbol{S}_{\text{in}}} = \Phi_{\boldsymbol{S}_{\text{out}}} (\Upsilon_{\boldsymbol{S}_{\text{in}}}^\top \Upsilon_{\boldsymbol{S}_{\text{in}}} + \lambda \boldsymbol{I}_{T'})^{-1} \Upsilon_{\boldsymbol{S}_{\text{in}}}^\top \tag{3}$$

$$\hat{\boldsymbol{C}}_{\boldsymbol{S}_{\text{tgt}}|\boldsymbol{S}_{\text{in}}} = \Phi_{\boldsymbol{S}_{\text{tgt}}} (\Upsilon_{\boldsymbol{S}_{\text{in}}}^\top \Upsilon_{\boldsymbol{S}_{\text{in}}} + \lambda \boldsymbol{I}_{T'})^{-1} \Upsilon_{\boldsymbol{S}_{\text{in}}}^\top \tag{4}$$

$$\Phi_{\boldsymbol{S}_{\text{out}}} = [\phi(\boldsymbol{s}_{\text{out}}(1)), \ldots, \phi(\boldsymbol{s}_{\text{out}}(T'))] \tag{5}$$

$$\Phi_{\boldsymbol{S}_{\text{tgt}}} = [\phi(\boldsymbol{s}_{\text{tgt}}(1)), \ldots, \phi(\boldsymbol{s}_{\text{tgt}}(T'))] \tag{6}$$

$$\Upsilon_{\boldsymbol{S}_{\text{in}}} = [\psi(\boldsymbol{s}_{\text{in}}(1)), \ldots, \psi(\boldsymbol{s}_{\text{in}}(T'))] \tag{7}$$

where $\phi(\boldsymbol{s}) = k(\boldsymbol{s}, \cdot)$ is the feature map for $\boldsymbol{S}_{\text{out}}$ and $\boldsymbol{S}_{\text{tgt}}$, and $\psi(\boldsymbol{s}) = h(\boldsymbol{s}, \cdot)$ is the feature map for $\boldsymbol{S}_{\text{in}}$. $k(\cdot)$ and $h(\cdot)$ are arbitrary positive definite kernel functions which compose a reproducing kernel Hilbert space (RKHS). We do not have to use the same kernel function for $k(\cdot)$ and $h(\cdot)$. $\boldsymbol{I}_{T'}$ is the T'-by-T' identity matrix, and λ is a regularization coefficient. Since $\phi(\boldsymbol{s})$ and $\psi(\boldsymbol{s})$ are the elements of RKHSs, we use the following relationship:

$$\boldsymbol{K}_{\boldsymbol{A},\boldsymbol{B}} = \Phi_{\boldsymbol{A}}^\top \Phi_{\boldsymbol{B}} \quad (\boldsymbol{A}, \boldsymbol{B} = \boldsymbol{S}_{\text{tgt}}, \boldsymbol{S}_{\text{out}}) \tag{8}$$

$$\boldsymbol{H}_{\boldsymbol{S}_{\text{in}}} = \Upsilon_{\boldsymbol{S}_{\text{in}}}^\top \Upsilon_{\boldsymbol{S}_{\text{in}}} \tag{9}$$

where the notation $\boldsymbol{K}_{\boldsymbol{A},\boldsymbol{B}}$ is the T'-by-T' Gram matrix between $\boldsymbol{A}$ and $\boldsymbol{B}$. For example, $\boldsymbol{K}_{\boldsymbol{S}_{\text{tgt}},\boldsymbol{S}_{\text{out}}}$ is the Gram matrix between $\boldsymbol{S}_{\text{tgt}}$ and $\boldsymbol{S}_{\text{out}}$, i.e., its (i, j)th element is $k(\boldsymbol{s}_{\text{tgt}}(i), \boldsymbol{s}_{\text{out}}(j))$. Similarly, $\boldsymbol{H}_{\boldsymbol{S}_{\text{in}}}$ is the Gram matrix for $\boldsymbol{S}_{\text{in}}$ and its (i, j)th element is $h(\boldsymbol{s}_{\text{in}}(i), \boldsymbol{s}_{\text{in}}(j))$. From the equations above, CMMD is estimated as follows:

$$\begin{aligned}\hat{L}(\boldsymbol{S}_{\text{in}}, \boldsymbol{S}_{\text{tgt}}, \boldsymbol{S}_{\text{out}}) = \frac{1}{T'^2} \{&\text{tr}(\boldsymbol{L}_{\boldsymbol{S}_{\text{in}}} \boldsymbol{K}_{\boldsymbol{S}_{\text{tgt}},\boldsymbol{S}_{\text{tgt}}}) \\ &+ \text{tr}(\boldsymbol{L}_{\boldsymbol{S}_{\text{in}}} \boldsymbol{K}_{\boldsymbol{S}_{\text{out}},\boldsymbol{S}_{\text{out}}}) \\ &- 2\text{tr}(\boldsymbol{L}_{\boldsymbol{S}_{\text{in}}} \boldsymbol{K}_{\boldsymbol{S}_{\text{tgt}},\boldsymbol{S}_{\text{out}}})\}\end{aligned} \tag{10}$$

$$\boldsymbol{L}_{\boldsymbol{S}_{\text{in}}} = \tilde{\boldsymbol{H}}_{\boldsymbol{S}_{\text{in}}}^{-1} \boldsymbol{H}_{\boldsymbol{S}_{\text{in}}} \tilde{\boldsymbol{H}}_{\boldsymbol{S}_{\text{in}}}^{-1} \tag{11}$$

$$\tilde{\boldsymbol{H}}_{\boldsymbol{S}_{\text{in}}} = \boldsymbol{H}_{\boldsymbol{S}_{\text{in}}} + \lambda \boldsymbol{I}_{T'}. \tag{12}$$

In the generation process, a random value is sampled from the modeled conditional distribution given input variables.

4. Proposed GMMN-Based Post-Filtering Method and Its Application to NDT

4.1 MS Extraction of Pitch Contour

The MS is the log-scaled power spectrum of a parameter sequence [17]. It represents the temporal structure of the sequence. The MS $\boldsymbol{S}_{\text{in}}$ of the input pitch sequence $\boldsymbol{y}_{\text{in}}$ is calculated using a short-time Fourier transform (STFT), as follows:

$$\boldsymbol{S}_{\text{in}} = [\boldsymbol{s}_{\text{in}}(1), \cdots, \boldsymbol{s}_{\text{in}}(\tau), \cdots, \boldsymbol{s}_{\text{in}}(T')] \tag{13}$$

$$\boldsymbol{s}_{\text{in}}(\tau) = [s_{\text{in}}(\tau, 0), \cdots, s_{\text{in}}(\tau, m), \cdots, s_{\text{in}}(\tau, M)]^\top \tag{14}$$

where τ is the segment index (one segment corresponds to one window of the STFT) and m is the modulation frequency index. The notation $s_{\text{in}}(\tau, m)$ is the MS of m at τth segment, T' is the total number of segments, and M is half the number of segments. The MS $\boldsymbol{S}_{\text{tgt}}$ of the target pitch sequence $\boldsymbol{y}_{\text{tgt}}$ is calculated similarly.

We used the zero-mean continuous F_0 sequence [17] to prevent errors caused by zero padding. After the post-filtering (described in Sects. 4.2 and 4.3), we can reconstruct the continuous F_0 sequence by carrying out an inverse STFT (ISTFT) using the filtered MS and the phase information of the input pitch contour. When calculating the MS, we need to use appropriate settings (e.g., windowing length) to achieve a perfect reconstruction through an STFT and ISTFT. We only use the lower modulation frequency MS (i.e., slowly changing components) for post-filtering because post-filtering the components with higher modulation frequencies causes unnatural temporal fluctuations in the F_0 sequence. This is reasonable because the LFO of ADT is equivalent to an addition operation in the lower modulation frequency.

4.2 Proposed Post-Filtering Method and NDT for Synthesized Singing Voices

We describe a GMMN-based post-filter built with our

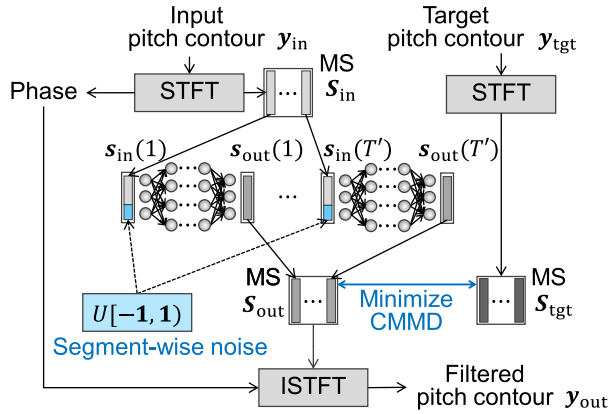


Fig. 3 Schematic diagram of post-filter built with proposed method.

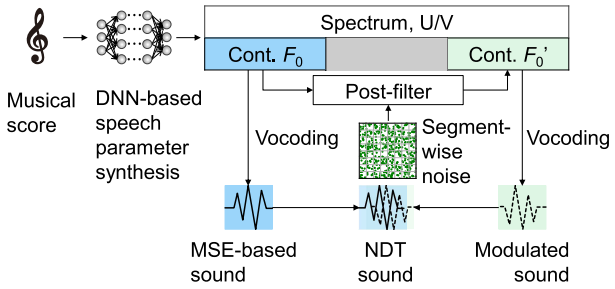


Fig. 4 Schematic diagram of NDT for synthesized singing voices. "Post-filter" in this figure corresponds to Fig. 3. In actual implementation, we decreased volume of modulated sound and added delay to it before mixing. "Spectrum" means spectral parameter and "U/V" means unvoiced/voiced label. "Cont. F_0" means continuous F_0 sequence.

method for randomly modulating the F_0 sequence of synthesized singing voices. Figure 3 represents the structure of the post-filter. We use a deterministically synthesized pitch contour as $\boldsymbol{y}_{\text{in}}$ and the natural pitch contour of the same song as $\boldsymbol{y}_{\text{tgt}}$.

In the training process, the conditional GMMN models the conditional distribution of natural continuous F_0. Let $\boldsymbol{n}(\tau) \sim U[-\mathbf{1}, \mathbf{1})$ be the prior noise vector at τth segment and $G(\cdot)$ be the conditional GMMN. The input of the conditional GMMN is the joint vector $[\boldsymbol{s}_{\text{in}}(\tau)^\top, \boldsymbol{n}(\tau)^\top]^\top$ and the output is the filtered MS $\boldsymbol{s}_{\text{out}}(\tau)$, i.e., $\boldsymbol{s}_{\text{out}}(\tau) = G([\boldsymbol{s}_{\text{in}}(\tau)^\top, \boldsymbol{n}(\tau)^\top]^\top)$. $\hat{L}(\boldsymbol{S}_{\text{in}}, \boldsymbol{S}_{\text{tgt}}, \boldsymbol{S}_{\text{out}})$ in Eq. (10) is minimized in training.

In the modulation process, we first calculate the MS $\boldsymbol{S}_{\text{in}}$ of an arbitrary input pitch sequence $\boldsymbol{y}_{\text{in}}$. Then, given the sequence of joint vectors $[\boldsymbol{s}_{\text{in}}(\tau)^\top, \boldsymbol{n}(\tau)^\top]^\top$, the conditional GMMN outputs a sequence of $\boldsymbol{s}_{\text{out}}(\tau)$. Using $\boldsymbol{S}_{\text{out}}$ and the phase information of $\boldsymbol{y}_{\text{in}}$, we then obtain the output pitch sequence $\boldsymbol{y}_{\text{out}}$ by carrying out an inverse STFT. By changing the noise, the post-filter can produce a different pitch contour. Finally, using the new pitch contour and the other original speech parameters, we obtain a randomly modulated voice by vocoding.

To conduct NDT on synthesized singing voices, we first generate a randomly modulated version of the input voice. The modulated voice is given some volume reduction and temporal delay to simulate the setting of ADT. Finally, we can obtain the layered voice by mixing the modulated voice with the original one. The schematic diagram is shown in Fig. 4.

4.3 Proposed Post-Filtering Method and NDT for Natural Singing Voices

A post-filter that modulates natural singing voices can be built with our method in a similar manner to that of the synthesized voices. The difference is the data that we use. We need a repeated singing voice database (i.e., the same singer singing the same songs multiple times). Let N be the number of recordings for each song, given one recording as the input, we can use the remaining $N-1$ recordings as the target. Thus, we can use ${}_N P_2 = N(N-1)$ pairs. Since the number of pairs increases by the squared order, we can easily increase the amount of training data. NDT can be used on natural singing voices in a similar manner to that of synthesized singing voices.

We consider two models. One is the singer-dependent (SD) model, which models the inter-utterance variation of a single singer. We can only use the singing voices of this singer as the input. The other is the singer-independent (SI) model, which models the singer-independent inter-utterance variation by using a database consisting of multiple singers. The SI model can be used for arbitrary singers.

4.4 Discussion

We now discuss the novelty and advantages of the proposed method. Our method involves randomness in the MS domain, not in the time domain. We developed this method for post-filtering, not generating. We previously built a GMMN-based frame-wise spectrum generation method for text-to-speech [1]. However, this method had two problems: 1) frame-wise noise vectors caused unnatural discontinuity in the output sequence and 2) the GMMN conditioned with linguistic features was incapable of providing perceptible variations because the linguistic features were sparse. The proposed method solves these problems. First, the MS of F_0 contours is a lower-dimensional representation that effectively captures segment-wise temporal structure, and filtering the components with lower modulation frequencies can randomly vary the F_0 contour without losing the continuity of the contour. Second, using a GMMN as a post-filter can avoid the sparseness problem and provide sufficient inter-utterance variations, as discussed in Sect. 5.2.

This is the first study to 1) provide natural inter-utterance pitch variations by using GMMNs to model MSs and 2) introduce such a post-filtering method for singing voices. Since this method takes into account the distribution of natural MSs, the variation in the post-filtered voices should be in the natural range. Figure 5 shows a pitch contour that was synthesized on the MSE basis and post-filtered (i.e., randomly modulated) pitch contours. We can see that our method samples different but continuous pitch contours. As mentioned in Sect. 1, inter-utterance variation enables

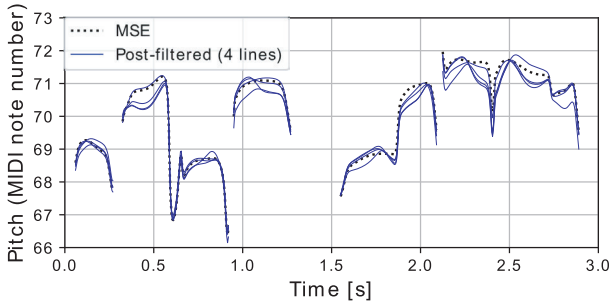


Fig. 5 Example of generated pitch contours. Four contours are sampled by our proposed method. Value of unity on vertical axis is equal to semitone.

one to choose a favorite from various ones. Since the pitch contour of each segment can be saved by fixing the input noise of that segment, it is possible to choose one's favorite pattern phrase-by-phrase and concatenate the segments to complete the whole song.

NDT is the first double-tracking method that uses a deep generative model. Conventional ADT involves a voice made by modulating the original voice deterministically without taking into account the natural distribution of pitch variations. NDT, on the other hand, takes into account the natural pitch variations, resulting in a more doubled-voice sense, as discussed in Sects. 5.4, 6.2, and 6.3.

We use data augmentation, which is an effective method for improving the accuracy of DNN modeling, in relation to the STFT. In an STFT, the calculated value significantly changes depending on the segment position (i.e., the frame index of the beginning of the segmentation). To cover such perturbations, we added all possible offsets to segment positions of the STFT analysis. This method augments the training data of the post-filter and presumably enhances its training accuracy.

We extend the framework so that natural singing voices can be input by creating a repeated singing voice database (Sect. 6.1). Although DT uses inter-utterance variations, mixing two voices with too much difference (especially note lengths) can cause unnaturalness. Since our method only changes the value of F_0, there is no worry about too much difference. Therefore, NDT can even be a safer and more practical choice than DT.

5. Experimental Evaluation Using Synthesized Singing Voices

5.1 Experimental Conditions

To evaluate our method applied to synthesized singing voices shown in Figs. 3 and 4, we used Japanese singing voice data of 31 songs from the HTS demo [21], 26 songs from the JSUT-song corpus [22], and 9 in-house songs. The singer of the second and third corpora was the same female and the singer of the first corpus was a different female. From these corpora, we used 58 songs (76 min and 50 s) for training the DNN for singing-voice synthesis, 28 songs (29 min and 41 s) of the HTS demo for training the post-filter, and 3 songs (1 min and 54 s) of the HTS demo (not included in the training data) for the evaluation. We used 3-fold data augmentation by transposing the context labels and voices up and down a semitone [10]. The sampling rate was 16 kHz. We used the WORLD analysis-synthesis system [23] to analyze and synthesize singing voices. The frame shift of the analysis-synthesis was set to 5 ms.

We predicted speech parameters on MSE basis using a feed-forward DNN consisting of a 705-dimensional input layer, 3 × 256-unit gated linear unit (GLU) [24] hidden layers, and a 127-unit linear output layer. The 705-dimensional input features consisted of 688-dimensional linguistic and musical features, a one-hot song code and a one-hot singer code [25]. The target and predicted speech parameter vector consisted of log-scaled continuous F_0, 40-dimensional mel-cepstral coefficients, band-aperiodicity [26], dynamic (delta- and delta-delta-) features [27] of those 42-dimensional parameters, and a binary unvoiced/voiced label. We used AdaGrad [28] to optimize DNN parameters. The number of training iterations was set to 50. The learning rate was 0.005, and the batch size was 500.

For the conditional GMMN, we used a feed-forward DNN consisting of an 11-dimensional input layer, 3 × 128-unit GLU hidden layers, and input-to-output residual net [29]. We used AdaGrad to optimize DNN parameters. The learning rate, minibatch size, and the number of training iterations were set to 0.005, 13,000, and 10, respectively. To calculate $\boldsymbol{L}_{\boldsymbol{S}_{\text{in}}}$ in Eq. (11), an approximation using 1024-dimensional random Fourier features [30] was used because calculating the inverse matrix was computationally infeasible. The input of the conditional GMMN was an 11-dimensional vector consisting of the first-order MS ($m = 1$) of the synthesized singing voice and a ten-dimensional noise vector generated from a uniform distribution $U[-\mathbf{1}, \mathbf{1})$. To stabilize the post-filter training, noise vectors were generated for each segment before training then fixed during training. The regularization coefficient λ was 0.01. We used Gaussian kernels for the input and output features, i.e., for the output, $\exp\{-\|\boldsymbol{s}_{\text{tgt}}(i) - \boldsymbol{s}_{\text{out}}(j)\|^2/\sigma^2\}$. We set σ for the input and output kernels to 100.0 and 1.0, respectively; these values were empirically chosen. The natural MS was normalized into the range $[0.01, 0.99]$. A 96-frame (480 ms) Hanning window and 48-frame (240 ms) segment shift were used for the STFT to extract MSs. To clarify the effect of pitch modulation, we used mel-cepstral coefficients, band-aperiodicity, and the unvoiced/voiced label of the corresponding natural singing voices in the vocoding process.

We conducted several subjective evaluations on 1) whether our method provided perceptible inter-utterance variation, 2) whether it degraded the naturalness of the pitch contours, and 3) whether the double-trackedness of NDT was higher than that of ADT. To make it easier for listeners to judge, we manually split the samples into segments in accordance with three conditions: short (one phrase), middle (twice as long as short or the same length as short, depend-

Table 1 Answer rate of perceived inter-utterance difference

Post-filtered	MSE	p-value
0.276	0.176	7.45×10^{-3}

Table 2 Naturalness and their p-values for middle- and long-duration singing voices

Length condition	Post-filtered	MSE	p-value
Middle	0.504	0.496	8.58×10^{-1}
Long	0.480	0.520	3.72×10^{-1}

ing on the phrases), and long (several phrases). The average lengths for the three conditions were 3.01, 4.88, and 10.24 s, respectively. We carried out the evaluations using the crowdsourcing platform "Lancers" [31] and gathered 25 participants for each evaluation.

5.2 Perception of Inter-Utterance Variation

To investigate whether the generated inter-utterance variation was perceptible for human listeners, we asked the participants whether they felt there was a difference between a pair of singing voices. We used short-duration voices to make it easier to remember subtle differences. We presented 20 pairs for each listener. Ten pairs were randomly post-filtered voices and the other ten pairs were identical MSE-based voices (control). Welch's t test was used to calculate the p-value.

Table 1 lists the results. For the MSE-based voices, the perception rate was 17.6% despite that the same voice was presented twice. On the other hand, the rate for the post-filtered voices was statistically significantly higher than that for the MSE-based voices. From this result, we can infer that perceptible inter-utterance variation can be produced using our method.

5.3 Naturalness of Post-Filtered Singing Voice

To determine whether our method degrades the quality of the pitch contour of synthesized voices, we presented ten pairs of post-filtered and MSE-based voices to the participants and asked them to choose the more natural one. We used the middle- and long-duration voices to make it easier to assess the overall naturalness.

Table 2 lists the results. We did not observe statistically significant difference for either the middle- or long-duration voices. This implies that our method did not reduce the naturalness of synthesized pitch contours.

5.4 Double-Trackedness of NDT

We evaluated the double-trackedness (i.e., the perceptual similarity to an actual double-tracked sound) of ADT and our NDT:

ADT: Modulating the log-scaled F_0 sequence by using an LFO and mixing its vocoded speech waveform with the MSE-based waveform. We used a sine wave oscillation whose depth and rate were 10% of a semitone and 0.775 Hz, respectively. The parameters were chosen by referring to a book on mixing [11]. The voices were modulated in the vocoder-parameter domain, not in the waveform domain because doing so is thought to produce fewer artifacts.

NDT: Modulating the log-scaled F_0 sequence using our method and mixing its vocoded speech waveform with the MSE-based waveform.

Table 3 Double-trackedness and their p-values for middle- and long-duration singing voices

Length condition	NDT	ADT	p-value
Middle	**0.724**	0.276	$< 10^{-10}$
Long	**0.736**	0.264	$< 10^{-10}$

With both methods, the modulated waveform was delayed by 20 ms and the volume decreased by 3 dB to produce the usual ADT setting [11]. We presented ten pairs of voices generated using the two methods above to the participants and asked them to choose the one that sounded more like an actual double-tracked sound. As discussed in Sect. 5.3, we used the middle- and long-duration samples.

Table 3 lists the results. Under both the middle and long conditions, the scores of our NDT were significantly higher than those of ADT. We can infer that our method achieves more double-trackedness than ADT does.

6. Experimental Evaluation Using Natural Singing Voices

6.1 Experimental Conditions

We created an in-house repeated singing voice database of 17 songs (13 min and 30 s). The songs were Japanese children's songs chosen from the demo of HTS [21]. Four male singers (A, B, C, D) sang all songs five times. They first listened to an example singing voice once with a headphone then sang along with the example and a metronome while looking at musical scores. Singing voices were recorded in an anechoic chamber. The sampling rate was 16 kHz. We used 14 songs (12 min and 6 s) for conditional GMMN training and 3 songs (1 min and 24 s) as the test data for post-filtering.

The overall setting of the post-filter is similar to that for synthesized singing voices. We now describe the different parts. We used the repeated singing voice database for the input and target of the conditional GMMN. We used WORLD [23] for the analysis and synthesis but used STRAIGHT [32] only for the extraction of F_0 because it was empirically more accurate for the voices of the database. In the vocoding process (analysis and synthesis), we used a 513-dimensional spectral envelope instead of mel-cepstral coefficients because it was not necessary to compress the dimension. The log-scaled F_0 was linearly interpolated in

the training, but we used only the voiced part in the objective evaluation. The architecture of the conditional GMMN and the training settings (the learning rate, the gradient, etc.) were the same as those presented in Sect. 5.1, except that the σ for the input and output kernels were 0.1, which was empirically chosen. We built both SD and SI models for all four singers. For the SI models, we used the singing voices of three other singers.

In both the subjective and objective evaluations, we compared the following four mixing methods:

ADT: The same as that presented in Sect. 5.4.

NDT (SD): Modulating the log-scaled F_0 sequence using our SD post-filter and mixing its vocoded speech waveform with the input voice.

NDT (SI): Modulating the log-scaled F_0 sequence using our SI post-filter and mixing its vocoded speech waveform with the input voice.

DT: Mixing the input voice with another recording of the same singer and same song as the input voice.

With ADT and NDT, the modulated voice was delayed by 20 ms. With all four methods, the volume of the secondary voice decreased by 3 dB.

6.2 Subjective Evaluation of NDT

To investigate the perceptual double-trackedness of our NDT, we conducted a listening test using Lancers [31]. We manually split the samples in accordance with two conditions: short (one phrase) and long (several phrases). The average lengths were 4.9 s and 10.5 s, respectively. One-hundred listeners participated for each condition. Each participant listened to 48 samples (3 samples $\times$ 4 singers $\times$ 4 methods) and scored the double-trackedness of the samples on a scale from 1 (the listener did not sense double-trackedness at all) to 5 (the listener sensed double-trackedness much).

Figures 6 and 7 show the mean opinion score (MOS) for each duration condition. Under both short and long conditions, the double-trackedness scores of ADT were low and those of NDT were close to those of DT. We conducted two-sided paired t-tests to determine whether there were differences between the MOS of NDT (SD) and NDT (SI). The p-values under the short and long conditions were 5.66×10^{-1} and 5.12×10^{-2}, respectively; thus, no statistically significant difference between SD and SI was observed when the significance level was 5%.

These results indicate that our NDT exhibits higher perceptual double-trackedness than ADT. We can also infer that it is possible to build singer-independent NDT models since the SI models are as effective as the SD ones. Some voice samples used in the evaluation are available online[†].

[†]https://sites.google.com/site/shinnosuketakamichi/research-topics/neural-double-tracking

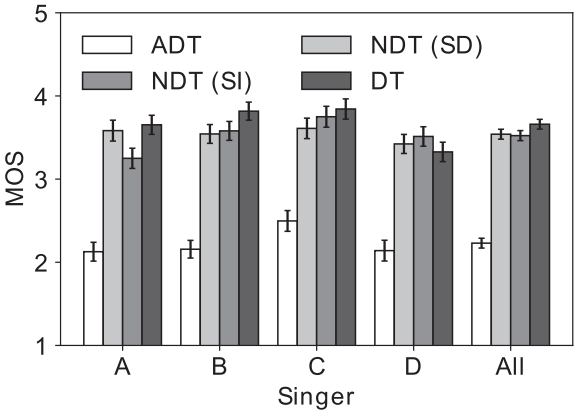


Fig. 6 MOS of double-trackedness for short condition.

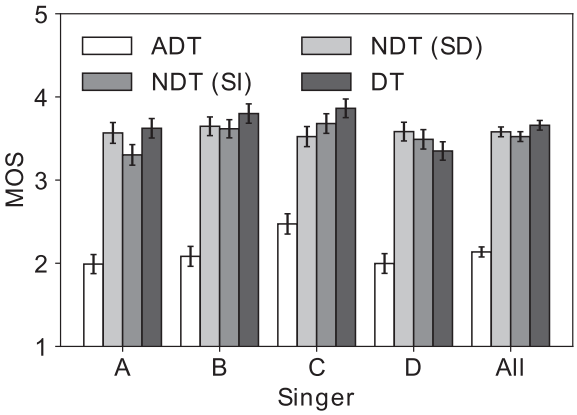


Fig. 7 MOS of double-trackedness for long condition.

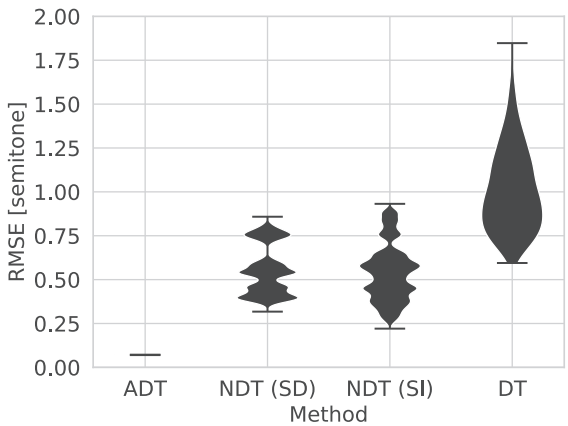


Fig. 8 Violin plot of RMSEs between pair of singing voices for 4 mixing methods.

6.3 Objective Evaluation of NDT

To investigate the numerical difference among the four mixing methods, we compared the root mean squared error (RMSE) between the log-scaled F_0 sequences of the input and secondary voices. For ADT, we used 272 samples (4 singers $\times$ 17 songs $\times$ 4 recordings). For NDT, we used

48000 samples (4 singers × 3 songs × 4 recordings × 1000 random noise patterns). For DT, we used 408 samples (4 singers × 17 songs × 6 pairs of recordings).

The distribution of RMSE is shown in Fig. 8. The RMSE of ADT is small and the variance of it is almost zero. On the other hand, both the RMSE and its variance of NDT are closer to those of DT than those of ADT. From this result, the high double-trackedness of NDT was objectively confirmed.

7. Conclusion

We proposed a GMMN-based post-filtering method that provides inter-utterance pitch variation to singing voices and discussed its application to a mixing method we developed called NDT. Inter-utterance variations of human singing not only enrich musical experiences but also enable DT. However, there are no such variations in deterministically synthesized voices and previously recorded voices. Our method uses a GMMN to model inter-utterance pitch variations among natural singing voices to randomly modulate singing voices as if the singer sang again. Experimental results indicate that 1) our method can generate pitch variations that are perceptible by human listeners without degrading the naturalness of the synthesized singing voices and that 2) higher double-trackedness can be achieved using our NDT than using ADT in both synthesized and natural singing voices.

In the future, we will extend our framework so that we can model inter-utterance variation of the duration and spectral parameters and combining it with that of the pitch to generate more natural variations.

Acknowledgments

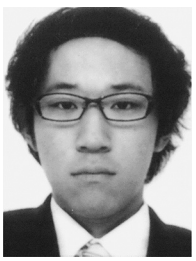
This work was supported by the SECOM Science and Technology Foundation.

References

[1] S. Takamichi, T. Koriyama, and H. Saruwatari, "Sampling-based speech parameter generation using moment-matching network," Proc. INTERSPEECH, pp.3961–3965, Stockholm, Sweden, Aug. 2017.
[2] R. Brice, Music Engineering, Elsevier Science, Oct. 2001.
[3] K. Womack, The Beatles Encyclopedia: Everything Fab Four, ABC-CLIO, June 2014.
[4] H. Kenmochi and H. Ohshita, "VOCALOID - commercial singing synthesizer based on sample concatenation," Proc. INTERSPEECH, pp.4011–4012, Antwerp, Belgium, Aug. 2007.
[5] M. Blaauw, J. Bonada, and R. Daido, "Data efficient voice cloning for neural singing synthesis," Proc. ICASSP, pp.6840–6844, Brighton, U.K., May 2019.
[6] D. Ayllón, F. Villavicencio, and P. Lanchantin, "A strategy for improved phone-level lyrics-to-audio alignment for speech-to-singing synthesis," Proc. INTERSPEECH, pp.2603–2607, Graz, Austria, Sept. 2019.
[7] K. Saino, H. Zen, Y. Nankaku, A. Lee, and K. Tokuda, "An HMM-based singing voice synthesis system," Proc. ICSLP, pp.2274–2277, Pittsburgh, U.S.A., Sept. 2006.
[8] K. Oura, A. Mase, T. Yamada, S. Muto, Y. Nankaku, and K. Tokuda, "Recent development of the HMM-based singing voice synthesis system - Sinsy," Proc. SSW7, pp.211–216, Kyoto, Japan, Sept. 2010.
[9] M. Nishimura, K. Hashimoto, K. Oura, Y. Nankaku, and K. Tokuda, "Singing voice synthesis based on deep neural networks," Proc. INTERSPEECH, pp.2478–2482, San Francisco, U.S.A., Sept. 2016.
[10] M. Blaauw and J. Bonada, "A neural parametric singing synthesizer modeling timbre and expression from natural songs," Applied Sciences, vol.7, no.12, Dec. 2017.
[11] R. Izhaki, Mixing Audio: Concepts, Practices, and Tools, Taylor & Francis, Oct. 2017.
[12] Y. Li, K. Swersky, and R. Zemel, "Generative moment matching networks," Proc. ICML, pp.1718–1727, Lille, France, July 2015.
[13] Y. Ren, J. Li, Y. Luo, and J. Zhu, "Conditional generative moment-matching networks," Proc. NIPS, pp.2928–2936, Barcelona, Spain, Dec. 2016.
[14] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. WardeFarley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," Proc. NIPS, pp.2672–2680, Montreal, Canada, Dec. 2014.
[15] D.P. Kingma and M. Welling, "Auto-encoding variational Bayes," arXiv, vol.abs/1312.6114, 2013.
[16] Y. Saito, Y. Ijima, K. Nishida, and S. Takamichi, "Non-parallel voice conversion using variational autoencoders conditioned by phonetic posteriorgrams and d-vectors," Proc. ICASSP, pp.5274–5278, Calgary, Canada, Apr. 2018.
[17] S. Takamichi, T. Toda, A.W. Black, G. Neubig, S. Sakti, and S. Nakamura, "Postfilters to modify the modulation spectrum for statistical parametric speech synthesis," IEEE Trans. Audio, Speech, Language Process., vol.24, no.4, pp.755–767, April 2016.
[18] H. Tamaru, Y. Saito, S. Takamichi, T. Koriyama, and H. Saruwatari, "Generative moment matching network-based random modulation post-filter for DNN-based singing voice synthesis and neural double-tracking," Proc. ICASSP, pp.7070–7074, Brighton, U.K., May 2019.
[19] K. Yu and S. Young, "Continuous F0 modeling for HMM based statistical parametric speech synthesis," IEEE Trans. Audio Speech Lang. Process., vol.19, no.5, pp.1071–1079, July 2011.
[20] L. Song, J. Huang, A. Smola, and K. Fukumizu, "Hilbert space embeddings of conditional distributions with applications to dynamical systems," Proc. ICML, pp.961–968, Montreal, Canada, June 2009.
[21] "HMM-based speech synthesis system (HTS)," http://hts.sp.nitech.ac.jp/.
[22] "JSUT-song," https://sites.google.com/site/shinnosuketakamichi/publication/jsut-song.
[23] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: A vocoder-based high-quality speech synthesis system for real-time applications," IEICE Trans. Inf. & Syst., vol.E99-D, no.7, pp.1877–1884, July 2016.
[24] Y.N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," arXiv, vol.abs/1612.08083, 2016.
[25] N. Hojo, Y. Ijima, and H. Mizuno, "DNN-based speech synthesis using speaker codes," IEICE Trans. Inf. & Syst., vol.E101-D, no.2, pp.462–472, 2018.
[26] M. Morise, "D4C, a band-aperiodicity estimator for high-quality speech synthesis," Speech Commun., vol.84, pp.57–65, 2016.
[27] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," Proc. ICASSP, Istanbul, Turkey, pp.1315–1318, June 2000.
[28] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," EURASIP Journal on Applied Signal Processing, vol.12, pp.2121–2159, July 2011.
[29] T. Kaneko, S. Takaki, H. Kameoka, and J. Yamagishi, "Generative adversarial network-based postfilter for STFT spectrograms," Proc. INTERSPEECH, pp.3389–3393, Stockholm, Sweden, Aug. 2017.
[30] A. Rahimi and B. Recht, "Random features for large-scale kernel

machines," Proc. NIPS, Vancouver, Canada, pp.1177–1184, Dec. 2008.
[31] "Lancers," https://www.lancers.jp/.
[32] H. Kawahara, J. Estill, and O. Fujimura, "Aperiodicity extraction and control using mixed mode excitation and group delay manipulation for a high quality speech analysis, modification and synthesis system STRAIGHT," MAVEBA, pp.1–6, Firentze, Italy, Sept. 2001.

Hiroki Tamaru received his B.E. degree in engineering from The University of Tokyo, Japan in 2018. He is studying for his M.S. degree in information physics and computing at The University of Tokyo. His research interests include singing voice synthesis, signal processing, and machine learning. He is a Student Member of ASJ and a Student Member of IEEE SPS.



Yuki Saito received his M.S. degree from the Graduate School of Information Science and Technology, The University of Tokyo, Japan in 2018. He is currently a Ph.D. student at The University of Tokyo. His research interests include speech synthesis, voice conversion, and machine learning. He has received eight paper awards including the 2017 IEICE ISS Young Researcher's Award in Speech Field. He is a Student Member of ASJ, IEEE SPS, and IEICE.

Shinnosuke Takamichi received his Ph.D. degree from the Graduate School of Information Science, Nara Institute of Science and Technology, Ikoma, Japan in 2016. He is currently an assistant professor at The University of Tokyo. He has received more than ten paper/achievement awards including the 3rd IEEE Signal Processing Society Japan Young Author Best Paper Award.

Tomoki Koriyama received his B.E. degree in computer science, and M.E. and Dr. Eng. degrees in information processing from Tokyo Institute of Technology, Tokyo, Japan in 2009, 2010, and 2013. In 2013, he joined the Research Laboratory of Interdisciplinary Graduate School of Science and Engineering, Tokyo Institute of Technology as a Japan Society for the Promotion of Science Research Fellow. He is currently an assistant professor at the Graduate School of Information Science and Technology, The University of Tokyo, Japan. Dr. Koriyama was a recipient of The Awaya Prize Young Researcher Award from Acoustic Society of Japan. He is a member of IEEE, ISCA, ASJ, IEICE, and IPSJ.

Hiroshi Saruwatari received his B.E., M.E., and Ph.D. degrees from Nagoya University, Japan in 1991, 1993, and 2000. He joined SECOM IS Laboratory, Japan in 1993, and Nara Institute of Science and Technology, Japan in 2000. He has been a professor at The University of Tokyo, Japan since 2014. His research interests include statistical audio signal processing, blind source separation (BSS), and speech enhancement. He has put his research into the world's first commercially available independent-component-analysis-based BSS microphone in 2007. He received paper awards from IEICE in 2001 and 2006, from TAF in 2004, 2009, 2012, and 2018, from IEEE IROS2005 in 2006, and from APSIPA in 2013 and 2018. He received the DOCOMO Mobile Science Award in 2011, Ichimura Award in 2013, The Commendation for Science and Technology by the Minister of Education in 2015, Achievement Award from IEICE in 2017, and Hoko-Award in 2018. He has been professionally involved in various volunteer work for IEEE, EURASIP, IEICE, and ASJ. He has been an APSIPA Distinguished Lecturer since 2018.