GENERATIVE MOMENT MATCHING NETWORK-BASED RANDOM MODULATION POST-FILTER FOR DNN-BASED SINGING VOICE SYNTHESIS AND NEURAL DOUBLE-TRACKING

Hiroki Tamaru[1], *Yuki Saito*[1], *Shinnosuke Takamichi*[1], *Tomoki Koriyama*[2], *and Hiroshi Saruwatari*[1]

[1] Graduate School of Information Science and Technology, The University of Tokyo, Japan.
[2] School of Engineering, Tokyo Institute of Technology, Japan.

ABSTRACT

This paper proposes a generative moment matching network (GMMN)-based post-filter that provides inter-utterance pitch variation for deep neural network (DNN)-based singing voice synthesis. The natural pitch variation of a human singing voice leads to a richer musical experience and is used in double-tracking, a recording method in which two performances of the same phrase are recorded and mixed to create a richer, layered sound. However, singing voices synthesized using conventional DNN-based methods never vary because the synthesis process is deterministic and only one waveform is synthesized from one musical score. To address this problem, we use a GMMN to model the variation of the modulation spectrum of the pitch contour of natural singing voices and add a randomized inter-utterance variation to the pitch contour generated by conventional DNN-based singing voice synthesis. Experimental evaluations suggest that 1) our approach can provide perceptible inter-utterance pitch variation while preserving speech quality. We extend our approach to double-tracking, and the evaluation demonstrates that 2) GMMN-based neural double-tracking is perceptually closer to natural double-tracking than conventional signal processing-based artificial double-tracking is.

***Index Terms*—** DNN-based singing voice synthesis, moment matching network, inter-utterance pitch variation, artificial double-tracking, modulation spectrum

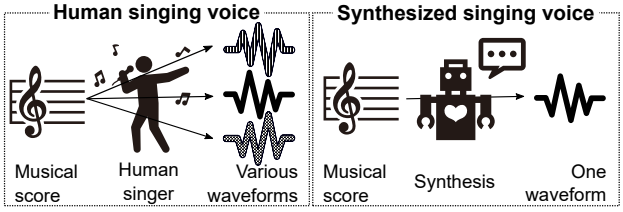

Fig. 1. Comparison of human and synthesized singing voices.

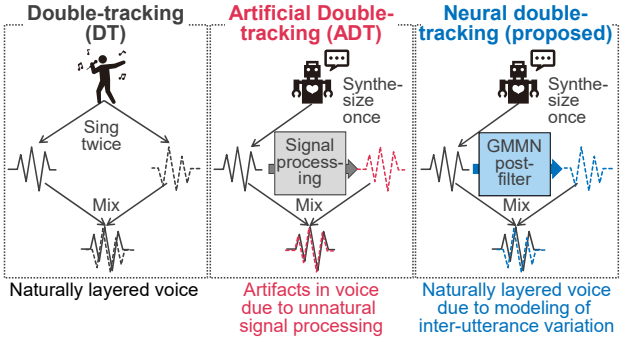

Fig. 2. Double-tracking (DT), artificial double-tracking (ADT), and proposed neural double-tracking. This figure shows ADT performed on synthesized voice, but it can also be performed on natural voice.

1. INTRODUCTION

These days, synthesized singing voices are being used for creating music. In particular, there are a variety of singing voice synthesis systems that work on the basis of unit selection synthesis (e.g., Vocaloid [1]), hidden Markov models (HMMs) [2, 3], and deep neural networks (DNNs) [4, 5]. One of the aims of these systems is to create expressive singing voices and music regardless of the users' gender or skill. Among them, DNN-based ones utilize a machine-learning-based synthesis process and have the potential to synthesize high-quality expressive voices.

However, the conventional DNN-based singing voice synthesis lacks *inter-utterance variation*, as shown in Fig. 1. Given one musical score, a human singer will sing it differently when asked to repeat it. The inter-utterance variation contributes to a richer musical experience. For instance, it proves that a singer is actually singing, not lip-synching. It also provides a music producer with the opportunity to choose a favorite from various recordings of the same song. In contrast, only one voice is synthesized from one musical score in conventional DNN-based singing voice synthesis because the synthesis process is deterministic. The lack of inter-utterance variation loses not only the rich musical experiences mentioned above, but also the ability to take advantage of *double-tracking* (DT) [6, 7] (Fig. 2). DT involves two or more vocal performances uttered by one human singer that are combined in a mix. It gives layeredness and richness to the resulting voices, thanks to the singer's inter-utterance variation. An alternative way is signal processing-based *artificial double-tracking* (ADT). Instead of recording multiple performances, ADT modulates one voice (e.g., through a delay and chorus effect) and mixes the original and modulated voices. Since ADT does not require multiple voices with inter-utterance variations, it can be easily applied to not only human voices but also synthesized voices. However, the effects of ADT are known to result in unnatural sound artifacts, such as tonal alterations and timbre coloration [8].

To address these problems, we propose a post-filter that gives inter-utterance pitch variations to a synthesized singing voice. Since such variations can be assumed to follow a certain complicated distribution, we use deep generative models, which are known to be capable of modeling complicated distributions. Building on our previous work on spectrum generation in text-to-speech synthesis [9], we utilize a generative moment matching network (GMMN) [10, 11] as the deep generative model because it is effective and easy to implement compared with other models. For example, generative ad-

versarial networks (GANs) [12] involve a difficult minimax problem and variational auto-encoders [13] are subject to the degration of the decoding quality due to over-regularization [14]. In our method, given the synthesized pitch contour and a prior noise vector, the conditional GMMN is trained to represent the distribution of natural pitch contours (i.e., the natural variation of the human singer's pitch contours). To capture the long-term pitch structure, the modulation spectrum (MS) [15] of the pitch contours is used, and the conditional GMMN models the natural MS variation. In the synthesis, given a prior noise vector, the GMMN randomly generates MSs that have natural inter-utterance variations, and the filtered pitch contour is created by modulating the input contour by the randomly generated MSs. In this study, we extend this framework to ADT (i.e., NDT: *neural double-tracking*). Since the GMMN-based post-filter provides natural inter-utterance variation, our NDT achieves naturally layered singing voices. The experimental evaluation demonstrates that our post-filtering approach can provide perceptible inter-utterance pitch variations while preserving speech quality and that the sound of our NDT is perceptually closer to that of natural DT than conventional signal processing-based ADT is.

2. CONVENTIONAL METHODS

2.1. DNN-based singing voice synthesis

In DNN-based singing voice synthesis [4], the relation between a musical score and the speech parameters of the parallel singing voice is modeled with DNNs. First, a musical score is converted into a sequence of vectors representing linguistic and musical contexts. Using DNNs, the parameters of the singing voices are predicted from the context sequence. Here, we minimize the mean square error (MSE) [4] between the natural and predicted speech parameters as follows:

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

where $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$ are natural and predicted speech parameter sequences lasting T frames, respectively. Since we are focusing on a 1-dimensional continuous fundamental frequency (F_0) [16], we 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 value at frame t and $\top$ is the transpose. In the synthesis, $\hat{\boldsymbol{y}} = [\hat{y}(1), \ldots, \hat{y}(t), \ldots, \hat{y}(T)]^\top$ is used as the speech parameter of the synthesized singing voice. Since this synthesis process is deterministic, the resulting sound has no inter-utterance variation.

2.2. ADT

Fig. 2 shows the difference between DT and ADT. DT is a recording method in which multiple performances of the same phrase are mixed in order to create a layered and rich sound [6, 7]. However, singing the same phrase twice in a similar fashion may be difficult and tiresome. ADT, which is an alternative method and only requires one recording, was originally 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 [6]. More recently, signal processing-based methods have been used; the most common one uses a chorus effect [8]. In the chorus effect, a waveform is copied and its pitch is modulated with a low-frequency oscillator; i.e., a sine wave is added to the original pitch contour, and the modulated sound is mixed with the original sound with some temporal difference to increase the doubled-voice feeling. Although such methods can be applied to DNN-based singing voice systems, they involve adding two signals with similar phases, which results in comb-filtering and its subsequent tonal alterations and timbre coloration [8].

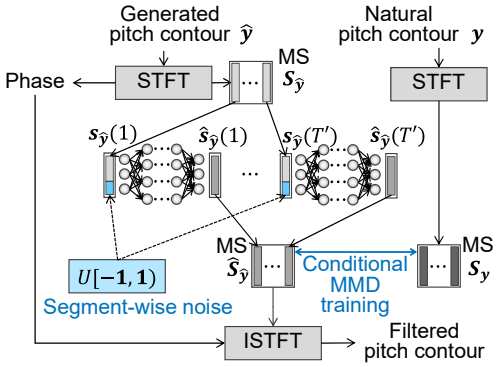

Fig. 3. Schematic diagram of our post-filter.

3. PROPOSED GMMN-BASED POST-FILTER AND ITS APPLICATION TO NDT

Here, we describe the proposed GMMN-based post-filter and how NDT is performed on MSE-based singing voices. First, the MSs of the log-scaled continuous F_0 of natural and generated pitch contours are extracted in order to capture the temporal structure of the contours. Second, A GMMN is trained to sample the MS randomly so that naturally varied pitch contours can be produced. Finally, NDT is performed using the non-filtered and filtered voices.

3.1. MS extraction

The MS is defined as the log-scaled power spectrum of a speech parameter sequence [15]. Namely, it captures temporal structures through the Fourier transform. We calculate the MS $\boldsymbol{S_y}$ of $\boldsymbol{y}$ through a short-time Fourier transform (STFT), as follows:

$$\boldsymbol{S_y} = [\boldsymbol{s_y}(1), \cdots, \boldsymbol{s_y}(\tau), \cdots, \boldsymbol{s_y}(T')], \tag{2}$$

$$\boldsymbol{s_y}(\tau) = [s_{\boldsymbol{y}}(\tau, 0), \cdots, s_{\boldsymbol{y}}(\tau, m), \cdots, s_{\boldsymbol{y}}(\tau, M)]^\top, \tag{3}$$

where τ is the segment index (one segment corresponds to one windowed continuous F_0 contour) and m is the modulation frequency index. $s_{\boldsymbol{y}}(\tau, m)$ indicates the MS of the modulation frequency m at segment τ. T' and M correspond to the total number of segments and one-half the number of segments of the STFT, respectively. $\boldsymbol{S_{\hat{y}}}$, the MS of $\hat{\boldsymbol{y}}$, is calculated in the same manner. Here, the zero-mean continuous F_0 sequence [15] is used to deal with errors caused by zero padding. We reconstruct the continuous F_0 sequence from the inverse STFT using the MS and original phase information and use the analysis settings (e.g., windowing length) to achieve a perfect reconstruction from the STFT and inverse STFT. We use only the lower modulation frequency components (i.e., components corresponding to slowly changing temporal structures) for post-filtering, because post-filtering the higher ones causes unnatural temporal fluctuations in the F_0 contours.

3.2. GMMN-based post-filtering

Here, we describe the GMMN-based post-filtering method to randomly modulate the generated continuous F_0 sequence $\hat{\boldsymbol{y}}$. Fig. 3 shows the schematic diagram of the post-filter. The GMMN [10] is a deep generative model, and it enables stabler training compared with a GAN [12]. The training criterion is called maximum mean discrepancy (MMD), which is a moment-based discrepancy between two distributions. The DNN takes a prior noise vector as input; this vector is the source of the inter-utterance variation in the post-filtering stage. In the training stage of the post-filter, given the MS of the generated continuous F_0 and segment-wise prior noise, the conditional distribution of the MS of natural continuous F_0 is modeled with the

DNN. Let $\boldsymbol{n}(\tau) \sim U[-\mathbf{1}, \mathbf{1})$ be the prior noise vector at segment τ, and $G(\cdot)$ be a DNN for post-filtering. The input of the DNN at segment τ is the joint vector $[\boldsymbol{s}_{\hat{\boldsymbol{y}}}(\tau)^\top, \boldsymbol{n}(\tau)^\top]^\top$, and the output is the filtered MS $\hat{\boldsymbol{s}}_{\hat{\boldsymbol{y}}}(\tau)$, i.e., $\hat{\boldsymbol{s}}_{\hat{\boldsymbol{y}}}(\tau) = G([\boldsymbol{s}_{\hat{\boldsymbol{y}}}(\tau)^\top, \boldsymbol{n}(\tau)^\top]^\top)$. Let $\hat{\boldsymbol{S}}_{\hat{\boldsymbol{y}}} = [\hat{\boldsymbol{s}}_{\hat{\boldsymbol{y}}}(1), \cdots, \hat{\boldsymbol{s}}_{\hat{\boldsymbol{y}}}(\tau), \cdots, \hat{\boldsymbol{s}}_{\hat{\boldsymbol{y}}}(T')]$; the following conditional MMD (CMMD) [11] is minimized in training:

$$
\begin{aligned}
L_{\mathrm{CMMD}}(\boldsymbol{S}_{\hat{\boldsymbol{y}}}, \boldsymbol{S}_{\boldsymbol{y}}, \hat{\boldsymbol{S}}_{\hat{\boldsymbol{y}}}) = & \frac{1}{T'^2}\{\mathrm{tr}(\boldsymbol{L}_{\boldsymbol{S}_{\hat{\boldsymbol{y}}}} \cdot \boldsymbol{K}_{\boldsymbol{S}_{\boldsymbol{y}}, \boldsymbol{S}_{\boldsymbol{y}}}) \\
& + \mathrm{tr}(\boldsymbol{L}_{\boldsymbol{S}_{\hat{\boldsymbol{y}}}} \cdot \boldsymbol{K}_{\hat{\boldsymbol{S}}_{\hat{\boldsymbol{y}}}, \hat{\boldsymbol{S}}_{\hat{\boldsymbol{y}}}}) \\
& - 2 \cdot \mathrm{tr}(\boldsymbol{L}_{\boldsymbol{S}_{\hat{\boldsymbol{y}}}} \cdot \boldsymbol{K}_{\boldsymbol{S}_{\boldsymbol{y}}, \hat{\boldsymbol{S}}_{\hat{\boldsymbol{y}}}})\}, \qquad (4)
\end{aligned}
$$

$$
\boldsymbol{L}_{\boldsymbol{S}_{\hat{\boldsymbol{y}}}} = \tilde{\boldsymbol{H}}_{\boldsymbol{S}_{\hat{\boldsymbol{y}}}}^{-1} \boldsymbol{H}_{\boldsymbol{S}_{\hat{\boldsymbol{y}}}} \tilde{\boldsymbol{H}}_{\boldsymbol{S}_{\hat{\boldsymbol{y}}}}^{-1}, \qquad (5)
$$

$$
\tilde{\boldsymbol{H}}_{\boldsymbol{S}_{\hat{\boldsymbol{y}}}} = \boldsymbol{H}_{\boldsymbol{S}_{\hat{\boldsymbol{y}}}} + \lambda \boldsymbol{I}_{T'}, \qquad (6)
$$

where $\boldsymbol{I}_{T'}$ is the T'-by-T' identity matrix and λ is a regularization coefficient. $\boldsymbol{K}_{\hat{\boldsymbol{S}}_{\hat{\boldsymbol{y}}}, \boldsymbol{S}_{\boldsymbol{y}}}$ is the T'-by-T' Gram matrix between $\hat{\boldsymbol{S}}_{\hat{\boldsymbol{y}}}$ and $\boldsymbol{S}_{\boldsymbol{y}}$; i.e., its i, jth component is $k(\hat{\boldsymbol{s}}_{\hat{\boldsymbol{y}}}(i), \boldsymbol{s}_{\boldsymbol{y}}(j))$, where $k(\cdot)$ is an arbitrary kernel function between two vectors. Similarly, $\boldsymbol{H}_{\boldsymbol{S}_{\hat{\boldsymbol{y}}}}$ is the Gram matrix for $\boldsymbol{S}_{\hat{\boldsymbol{y}}}$; i.e., its i, jth component is $h(\boldsymbol{s}_{\hat{\boldsymbol{y}}}(i), \boldsymbol{s}_{\hat{\boldsymbol{y}}}(j))$, where $h(\cdot)$ is an arbitrary kernel function between two vectors. Note that we can choose different kernel functions between $k(\cdot)$ and $h(\cdot)$. After training, the model represents the natural MS distribution (i.e., variation of the pitch contours) given the generated MS. The synthesis stage first generates $\hat{\boldsymbol{y}}$ from DNN-based singing voice synthesis and calculates its MS and phase. It then filters the MS, for it to have natural variation, by using the trained GMMN and randomly sampled prior noise. The final F_0 contour is generated by performing the inverse STFT on the filtered MS and the non-filtered phase.

3.3. Application to NDT

Here, we describe how to give natural layeredness to a synthesized singing voice. After the speech parameter generation, one waveform is synthesized in the standard vocoding process. Another waveform is synthesized using F_0 values modulated by our post-filter. The final double-tracked voice is obtained by mixing the modulated waveform with the non-filtered one with some temporal difference.

3.4. Discussion

In [9], we built a GMMN-based spectrum generator by using frame-wise noise vectors in text-to-speech synthesis. However, the method had two problems: 1) such frame-wise noise and variation modeling caused unpleasant sounds in the case of F_0 contours and 2) the GMMN conditioned with linguistic features could not provide perceptible variations because of the sparseness of the linguistic features. This paper's method effectively solves these problems. This is because the MS of F_0 contours is a lower-dimensional and effective representation with which to capture segment-wise temporal structure, and filtering the lower modulation frequency components can modulate the F_0 contour and at the same time preserve the continuity of the contour. The other reason is that the use of GMMNs as a post-filter can avoid the sparseness problem and can provide perceptible inter-utterance variations, as shown in Section 4.2.

Since our method considers the distribution of natural MSs, the variation of the post-filtered voices should be in the natural range. Fig. 4 shows MSE-based (i.e., deterministic and non-filtered) and post-filtered (i.e., randomly modulated) pitch contours. We can see that our method samples random, but continuous pitch contours. This is the first study to 1) provide natural inter-utterance pitch variations by using GMMNs to model MSs and to 2) introduce such a system as a post-filter for singing voices. As described in Section 1, inter-utterance variation enables one to pick a favorite from various voices. Since the pitch contour of each segment can be saved by fixing the input noise, users can choose their favorite contour phrase-by-phrase and concatenate phrases to make the whole song.

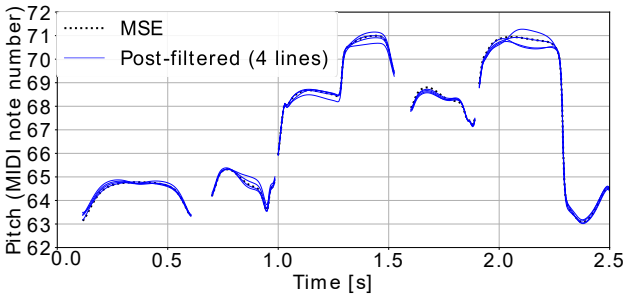

Fig. 4. Example of generated pitch contours. We used proposed method to sample four contours. Value of unity on vertical axis is equal to semitone.

Conventional ADT involves a waveform made by modulating the original sound deterministically without considering the natural distribution of pitch variations. NDT is a new approach in that the sound is modulated considering the natural pitch variations and that it results in more doubled-voice feeling as shown in Section 4.4.

Data augmentation is a powerful method for improving the accuracy of DNN modeling. In this study, we utilized data augmentation in relation to the STFT. Since the MS utilizes an STFT, the value changes significantly with the segment position (i.e., the frame index of the beginning of the segmentation). To cover such perturbations, we added all possible offsets to the first frame of the STFT analysis. This method effectively augments the training data of our post-filter and improves its training accuracy.

Since our method utilizes pitch contours as the input and the output of the neural network, we expect that it can be extended to NDT for human singers, i.e., where a human singer sings only once and the extended NDT synthesizes a naturally layered voice. We will pursue this idea in the future (see Section 5).

4. EXPERIMENTAL EVALUATION

4.1. Experimental conditions

We used Japanese singing voice data of 31 songs from a singing-voice-synthesis demo of HTS [17], 26 songs from the JSUT-song corpus [18], and nine in-house songs sung by the same singer as in the JSUT-song corpus. From these corpora, we used 58 songs for training the DNN for singing voice synthesis, 28 songs of the HTS demo for training our post-filter, and three songs of the HTS demo that were not included in the training data for the evaluation. Label data were augmented 3-fold by transposing the songs up and down a semitone [5]. The speech signals were sampled at a rate of 16 kHz. The WORLD analysis-synthesis system [19] was used to extract the speech features and to synthesize the waveforms. The frame shift was 5 ms. The DNN used to predict the MSE-based speech parameters was a feed-forward network including a 705-dimensional input layer, 3×256-unit gated linear unit (GLU) [20] hidden layers, and a 127-unit linear output layer. The learning rate was 0.005, and the batch size was 500. The learning process was repeated for 50 epochs by using AdaGrad [21]. The 705-dimensional input features included 688-dimensional linguistic and musical features, a one-hot song code and a one-hot singer code [22]. The DNN predicted a 127-dimensional vector consisting of 40-dimensional mel-cepstral coefficients, log-scaled continuous F_0, band-aperiodicity [23, 24], dynamic (delta- and delta-delta-) features [25] of those 42-dimensional parameters, and a binary unvoiced/voiced label. The DNN used

Table 1. Answer rate of perceived inter-utterance difference

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

in the conditional GMMN was a feed-forward network including an 11-dimensional input layer, 3×128-unit GLU hidden layers, and input-to-output residual net [26]. The learning process was repeated ten times by using AdaGrad. The learning rate was 0.005, and the batch size was 13000. We used an approximation using 1024-dimensional random Fourier features [27] to calculate $\boldsymbol{L}_{\boldsymbol{S}_{\hat{\boldsymbol{y}}}}$ in Eq. (5) because calculating the inverse matrix was computationally infeasible. The 11-dimensional input vector consisted of the first-order MS ($m = 1$) of the singing voice synthesized based on MSE and a ten-dimensional noise vector generated from a uniform distribution $U[-\mathbf{1}, \mathbf{1})$. For stable training, noise vectors were first generated for each segment and were then fixed during training. The regularization coefficient λ was set to 0.01. The kernel function was Gaussian, i.e., $\exp\{-||\boldsymbol{s}_{\boldsymbol{y}}(i) - \hat{\boldsymbol{s}}_{\hat{\boldsymbol{y}}}(j)||^2/\sigma^2\}$. A Gaussian kernel was used for the input features as well. The σ for the input features was 100.0, and the σ for the output features was 1.0; these values were empirically chosen. The natural MS was normalized so that all data fell within the range $[0.01, 0.99]$. For the STFT, a 96-frame (480 ms) Hanning window and 48-frame (240 ms) segment shift were used. To clarify the effect of pitch modulation, we used spectral parameters, band-aperiodicity, and the unvoiced/voiced label of the natural singing voices in the vocoding process.

We conducted three subjective evaluations for determining 1) whether our post-filter provided perceptible inter-utterance variations, 2) whether the post-filter degraded the naturalness of the synthesized voices, and 3) whether the sound of NDT was perceptually closer to that of natural DT than ADT was. The evaluations were performed using the Lancers crowdsourcing platform [28]. To make it easier for listeners to judge, we manually split the songs into segments in correspondence with three conditions: short (one phrase), middle, and long (several phrases). The average lengths for the three conditions were 3.01 s, 4.88 s, and 10.24 s, respectively.

4.2. Perception of inter-utterance pitch variation

To determine whether the inter-utterance pitch variation could be perceived by human listeners, we asked 25 listeners whether they felt there was a difference between a pair of singing voices. The short-duration voices were used because it is easier to remember subtle differences if shorter sounds are presented. Each participant listened to 20 pairs of singing voices consisting of ten pairs of randomly post-filtered voices (proposed) and ten pairs of identical MSE-based voices. We used Welch's *t* test to calculate the *p*-value.

Table 1 shows the results. The perception rate of MSE was 17.6% despite that the same voices were presented. On the other hand, the rate of our method was statistically significantly higher than that of MSE. This suggests that our method is capable of producing perceptible inter-utterance variations.

4.3. Naturalness of post-filtered voice

We tried to determine whether our post-filter degraded the quality of the synthesized voices. We asked 25 participants to listen to ten pairs of post-filtered and MSE-based voices and to choose the more natural one. The middle- and long-duration voices were used to make it easy to judge the overall naturalness.

Table 2 shows the results. There was no statistically significant difference for either the middle-duration voices or the long ones. This implies that the post-filter did not degrade the naturalness of the singing voices.

Table 2. Preference scores of singing voice naturalness and their *p*-values for the middle and long conditions

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

Table 3. Preference scores of double-trackedness and their *p*-values for the middle and long conditions

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

4.4. Evaluation of NDT

We evaluated the *double-trackedness* (i.e., the perceptual similarity to naturally double-tracked sound) of the proposed NDT and conventional ADT:

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

ADT: Modulating the log-scaled F_0 sequence by using a low-frequency oscillator and mixing its vocoded speech waveform with the MSE-based waveform. The shape, rate, and depth of the oscillation were a sine wave, 0.775 Hz, and 10% of a semitone, respectively. The parameters were determined by referring to [8]. The modulation was performed in the vocoder parameter domain, not in the waveform domain, because doing so is considered to produce fewer artifacts.

In both methods, the modulated wave was delayed by 20 ms and the volume was reduced by 3 dB to produce the usual ADT setting [8]. We asked 25 participants to listen to ten pairs of sounds generated under the two conditions above and to choose the one that sounded more like a naturally double-tracked sound. As before, the middle- and long-duration voices were used.

Table 3 shows the results. The score of our NDT was significantly higher than that of conventional ADT for both the middle and long conditions. This suggests that our deep-generative method gives a more double-tracked feeling than the conventional signal processing-based approach does.

5. CONCLUSION

This paper described a GMMN-based post-filter for generating inter-utterance pitch variation and its application to ADT. Although a human singer never sings the same song in the same way twice, conventional singing voice synthesis systems only generate a single waveform that is considered to be the most natural one. Our post-filter models inter-utterance pitch variation by using a GMMN to match the statistical moments of the MS of synthesized voices and those of natural voices. Experimental results suggest that the post-filter can generate pitch variations that are perceptible by human listeners without degrading the naturalness of the synthesized singing voices. We also discussed the effectiveness of applying the post-filter to ADT. Experimental results suggest that the proposed NDT is perceptually closer to natural DT than conventional ADT is.

Future work will include modeling the inter-utterance variation of the duration and spectral parameters and combining them with that of pitch in order to reproduce more natural variation. Another topic is formulating a post-filter that can be used to modulate the MS of natural singing voices.

Acknowledgements: Part of this work was supported by SECOM Science and Technology Foundation.

6. REFERENCES

[1] H. Kenmochi and H. Ohshita, "Vocaloid - commercial singing synthesizer based on sample concatenation," in *Proc. INTERSPEECH*, Antwerp, Belgium, Aug. 2007, pp. 4011–4012.

[2] K. Saino, H. Zen, Y. Nankaku, A. Lee, and K. Tokuda, "An HMM-based singing voice synthesis system," in *Proc. ICSLP*, Pittsburgh, U.S.A., Sep. 2006, pp. 2274–2277.

[3] K. Oura, A. Mase, T. Yamada, S. Muto, Y. Nankaku, and K. Tokuda, "Recent development of the HMM-based singing voice synthesis system - Sinsy," in *Proc. SSW7*, Kyoto, Japan, Sep. 2010, pp. 211–216.

[4] M. Nishimura, K. Hashimoto, K. Oura, Y. Nankaku, and K. Tokuda, "Singing voice synthesis based on deep neural networks," in *Proc. INTERSPEECH*, San Francisco, U.S.A., Sep. 2016, pp. 2478–2482.

[5] 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.

[6] R. Brice, *Music Engineering*, Elsevier Science, Oct. 2001.

[7] K. Womack, *The Beatles Encyclopedia: Everything Fab Four*, ABC-CLIO, Jun. 2014.

[8] R. Izhaki, *Mixing Audio: Concepts, Practices, and Tools*, Taylor & Francis, Oct. 2017.

[9] S. Takamichi, K. Tomoki, and H. Saruwatari, "Sampling-based speech parameter generation using moment-matching network," in *Proc. INTERSPEECH*, Stockholm, Sweden, Aug. 2017, pp. 3961–3965.

[10] Y. Li, K. Swersky, and R. Zemel, "Generative moment matching networks," in *Proc. ICML*, Lille, France, Jul. 2015, pp. 1718–1727.

[11] Y. Ren, J. Li, Y. Luo, and J. Zhu, "Conditional generative moment-matching networks," in *Proc. NIPS*, Barcelona, Spain, Dec. 2016, pp. 2928–2936.

[12] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. NIPS*, Montreal, Canada, Dec. 2014, pp. 2672–2680.

[13] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv*, vol. abs/1312.6114, 2013.

[14] Y. Saito, Y. Ijima, K. Nishida, and S. Takamichi, "Non-parallel voice conversion using variational autoencoders conditioned by phonetic posteriorgrams and d-vectors," in *Proc. ICASSP*, Calgary, Canada, Apr. 2018, pp. 5274–5278.

[15] 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 Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 4, pp. 755–767, Apr. 2016.

[16] K. Yu and S. Young, "Continuous F0 modeling for HMM based statistical parametric speech synthesis," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 5, pp. 1071–1079, Jul. 2011.

[17] "HMM-based speech synthesis system (HTS)," http://hts.sp.nitech.ac.jp/.

[18] "JSUT-song," https://sites.google.com/site/shinnosuketakamichi/publication/jsut-song.

[19] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: a vocoder-based high-quality speech synthesis system for real-time applications," *IEICE transactions on information and systems*, vol. E99-D, no. 7, pp. 1877–1884, Jul. 2016.

[20] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," *arXiv*, vol. abs/1612.08083, 2016.

[21] 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, Jul. 2011.

[22] N. Hojo, Y. Ijima, and H. Mizuno, "An investigation of DNN-based speech synthesis using speaker codes," in *Proc. INTERSPEECH*, San Francisco, U.S.A., Sep. 2016, pp. 2278–2282.

[23] H. Kawahara, Jo 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," in *MAVEBA*, Firentze, Italy, Sep. 2001, pp. 1–6.

[24] Y. Ohtani, T. Toda, H. Saruwatari, and K. Shikano, "Maximum likelihood voice conversion based on GMM with STRAIGHT mixed excitation," in *Proc. INTERSPEECH*, Pittsburgh, U.S.A., Sep. 2006, pp. 2266–2269.

[25] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in *Proc. ICASSP*, Istanbul, Turkey, Jun. 2000, pp. 1315–1318.

[26] T. Kaneko, S. Takaki, H. Kameoka, and J. Yamagishi, "Generative adversarial network-based postfilter for STFT spectrograms," in *Proc. INTERSPEECH*, Stockholm, Sweden, Aug. 2017, pp. 3389–3393.

[27] A. Rahimi and B. Recht, "Random features for large-scale kernel machines," in *Proc. NIPS*, Vancouver, Canada, Dec. 2008, pp. 1177–1184.

[28] "Lancers," https://www.lancers.jp/.