Voice Factor Control Using FIR-Based Fast Neural Vocoder for Speech Generation Applications

Yamato Ohtani[1], Takuma Okamoto[1], Tomoki Toda[2,1], Hisashi Kawai[1]
[1]*National Institute of Information and Communications Technology, Japan,* [2]*Nagoya University, Japan*
{yamato.ohtani, okamoto, hisashi.kawai}@nict.go.jp, tomoki@icts.nagoya-u.ac.jp

***Abstract*—We have proposed a fast neural vocoder based on the source-filter model introducing finite impulse response (FIR) filters called FIRNet. FIRNet is highly compatible with digital signal processing (DSP) and can, therefore, generate waveforms from vocoder parameters and modified voice factors, such as tone, intonation, and timbre, using DSP. Although modern neural waveform generation systems, such as voice conversion and text-to-speech (TTS), have been able to generate human-like synthetic speech and imitate the reference speaker's timbre, it is challenging for these systems to manually control arbitrary voice factors, unlike traditional TTS systems. By applying FIRNet to modern neural waveform generation systems, they can achieve arbitrary voice factor controllability. We will demonstrate two applications using FIRNet with DSP-based voice factor controls: one is analysis-synthesis, and the other is text-to-speech.**

***Index Terms*—Speech synthesis, neural vocoder, voice factor control, digital signal processing.**

I. Introduction

Recent advances in deep learning technology have enabled high-fidelity speech generation applications, including text-to-speech (TTS) [1]–[4] and voice conversion (VC) [5]–[7]. The several state-of-the-art neural speech generation systems can control target voice qualities and speaking styles by using reference speech data [8]–[10], pitch shifting [11]–[13], and prompting [14]. However, these systems have some limitations in terms of controllability, such as the need for references, narrow pitch control ranges, and iterative trials to achieve the desired voice qualities. In practical use, it is important factors for many users of speech generation systems to control voice factors such as timbre (depth and hoarseness) and prosody (tone and intonation) freely and indeed to create various contents using synthetic speech without reference speech. Reviewing the traditional speech generation systems such as parametric TTS systems [15], [16], they can control them by direct modifications of the vocoder parameters [17] based on the linear digital signal processing (DSP) and generate speech waveforms using the synthesizer based on the source-filter model (SFM) [18].

To achieve similar voice factor controls to those of modern speech generation systems, we prototyped neural speech generation systems using the FIRNet [19], a high-speed SFM-based neural vocoder with time-variant finite impulse response (FIR) filters. Potentially, SFM-based neural vocoders can convert modified vocoder parameters into their corresponding waveforms exactly because the parameter modifications involve linear processing, and the above neural vocoders can

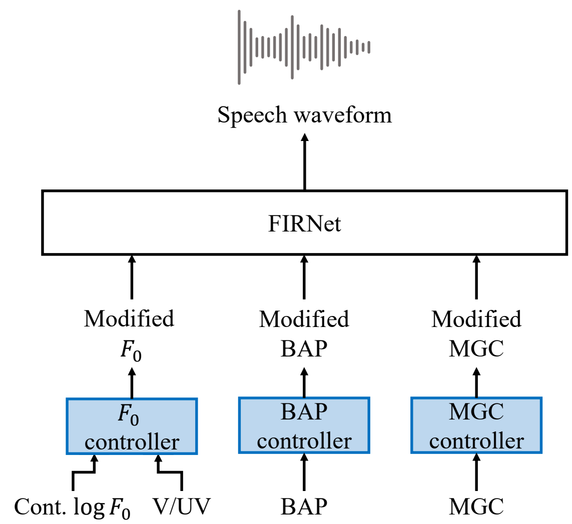

Fig. 1. Details of FIRNet with DSP-based parameter controllers.

capture their characteristics. In addition, the FIRNet performs higher speed than other SFM-based neural vocoders reported in [19]. Thus, it is effective to use the FIRNet in terms of the achievement of our purpose. In the demonstration, we present two types of applications: one is speech modification using analysis-synthesis, and the other is a TTS system.

II. FIRNet with DSP-based Voice Factor Controllers

In the demonstration system for analysis-synthesis, we combine the FIRNet with three DSP-based parameter controllers, as shown in Fig. 1. Note that we employ the parallel FIRNet architecture due to the high inference speed and speech quality of synthetic speech [19]. As vocoder parameters, the FIRNet uses F_0, V/UV, BAP, and MGC that mean fundamental frequency, voiced/unvoiced flag, banded aperiodicity [20] and mel-generalized cepstrum [21], respectively. In this paper, these vocoder parameters are extracted with WORLD analysis tools [17], [22], [23] and Speech Signal Processing Toolkit (SPTK) [24].

A. F_0 controller

The F_0 controller generates the modified F_0 contour based on the input control parameters as follows:

$$\tilde{f}_t = r * \exp\{v * (\log f_t - m) + m\} * u_t, \tag{1}$$

where f_t $\tilde{f}_t$, u_t, r, v and m denote input continuous F_0 at the tth frame, modified F_0, V/UV parameter, the control parameter for F_0 shifting rate, that for the variation of the F_0 and the average of input continuous $\log F_0$, respectively. Note that the lower values of r and v are limited to 0.

B. BAP controller

BAP controller can control hoarseness of speech by changing the BAP vector $\mathbf{b}_t$ according to the user instructions as follows:

$$\tilde{\mathbf{b}}_t = \begin{cases} \min\left(\mathbf{b}_u, (1-c)\mathbf{b}_u + c\mathbf{b}_t\right) & \text{if } 0 \leq c < 1 \\ \max\left(\mathbf{b}_l, (c-1)\mathbf{b}_l + (2-c)\mathbf{b}_t\right) & \text{else if } 1 \leq c, \end{cases} \tag{2}$$

where $\mathbf{b}_t$, $\mathbf{b}_u$, $\mathbf{b}_l$ and c denote the linear-scaled BAP whose range is 0 (periodic) to 1 (aperiodic), the upper-value vector of the BAP, the lower value vector of the BAP, and BAP control parameter whose range is 0 to 2, respectively. We design this system so that c closer to 0 results in aperiodic speech, and c closer to 2 results in periodic speech.

C. MGC controller

The MGC controller applies the frequency warping function [21], [25] to the MGC. In this system, we use pysptk[1] for the frequency warping function, which is a Python wrapper of SPTK.

III. Experiments

To check whether modified vocoder parameters are converted into their corresponding synthetic waveform using the FIRNet, we have conducted an objective speaker similarity test using the speaker similarity assessment model in the analysis-synthesis task. In experiments, we employed English male and female speakers in Hi-Fi-Captain corpus [26] for training and evaluation of the FIRNet. For building the FIRNet models, we resampled these waveforms from 48 kHz to 24 kHz. To accelerate inference speed, we set the hop size and the number of filter coefficients to 240 and 384, respectively. In this evaluation, focused on MGC modification using the frequency warping function, we calculated speaker similarity scores between synthetic waveforms of the FIRNet and those of the WORLD synthesizer using VoxSim[2] [27]. This is because the WORLD synthesizer is based on DSP and can generate waveforms corresponding to the modified vocoder parameters correctly. We set four warping parameters: -0.2, -0.1, 0.0, 0.1, and 0.2.

Figure 2 shows the heatmap of the objective speaker similarity scores. In the same warping parameters of the WORLD synthesizer and the FIRNet, objective speaker similarity scores are higher than other warping parameter conditions. This indicates that the FIRNet can accurately reflect warping parameter information in synthetic waveforms, just like the WORLD synthesizer. Consequently, by applying FIRNet to neural waveform generation applications such as VC and TTS, they can achieve voice factor control abilities that are independent of the training data.

[1]https://github.com/r9y9/pysptk

[2]https://huggingface.co/spaces/junseok520/VoxSIM

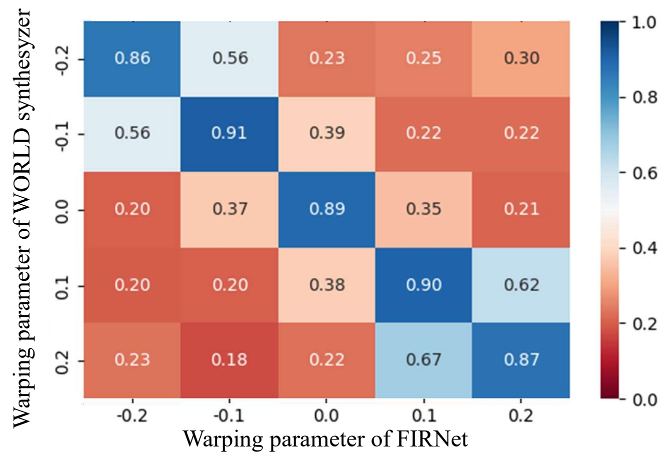

Fig. 2. Heatmap of the objective speaker similarity scores. Note that higher scores mean higher speaker similarities.

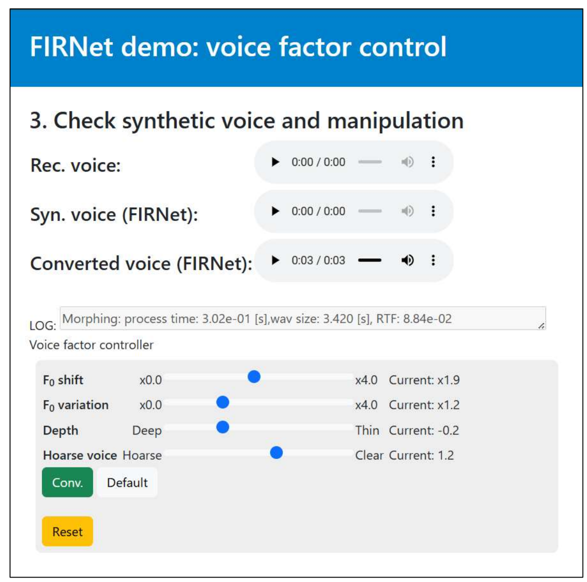

Fig. 3. GUI for analysis-synthesis application.

IV. Demonstration Applications

During the demonstration session, we will present two types of voice factor control applications: analysis-synthesis (AS) and neural text-to-speech (TTS). Each application can run rapidly on the laptop PC. Their graphical user interfaces (GUIs) are shown as Figs 3 and 4. These GUIs are designed to allow even non-expert users to intuitively control voice factors.

A. Analysis-synthesis application

In the AS application, we directly utilize the FIRNet with DSP-based voice factor controllers, as described in Sec II. This application is an interactive demonstration and has five steps: 1) users record their speech, 2) users check the recording

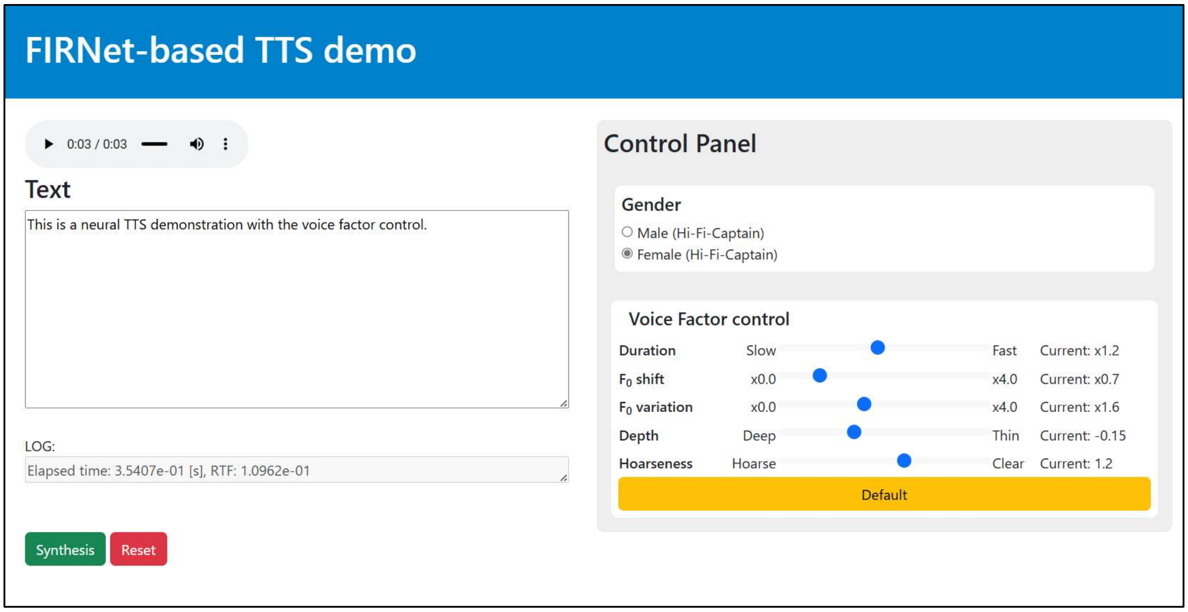

Fig. 4. GUI for neural TTS application.

condition, 3) this system extracts vocoder parameters from users' recorded speech, 4) reconstructs users' speech, and 5) users control voice factors (F_0 shift and variation, depth (MGC control), and hoarseness (BAP control)) through the control panel and generate converted synthetic speech. In this application, the real-time factors (RTFs) of speech analysis, generations of re-synthetic speech and converted speech are around 0.05, 0.085 and 0.09, respectively, on the single CPU (Intel(R) Core(TM) i7-1255U 1.70 GHz).

B. Neural TTS application

In the neural TTS application, we combine the FIRNet with the neural acoustic model as shown in Fig 5, which refers to the acoustic model used in Mobile PresenTra [28], which is one of the non-autoregressive models with monotonic alignment search [29]. This acoustic model utilizes a transformer-based encoder and decoder based on the ConvNeXt architecture [30]. Unlike the acoustic model used in Mobile PresenTra, this acoustic model outputs vocoder parameters, such as MGC, continuous $\log F_0$, V/UV, and BAP, instead of mel-spectrograms, and removes the pitch predictor and the energy predictor. Additionally, this model introduces duration, F_0, BAP, and MGC controllers, which modify their corresponding vocoder parameters based on the respective control parameters. The GUI of the neural TTS system comprises a text area, a gender selector, and a voice factor control panel. Compared to the AS application, users can also control duration through the voice factor control panel. Users input text to synthesize, select gender, and set each control parameter, and then they can obtain desired synthetic waveforms freely. The RTF of this application is around 0.12 (acoustic model: 0.02, parameter modification: 0.005 and FIRNet: 0.085) on the same single CPU condition as Sec IV-A.

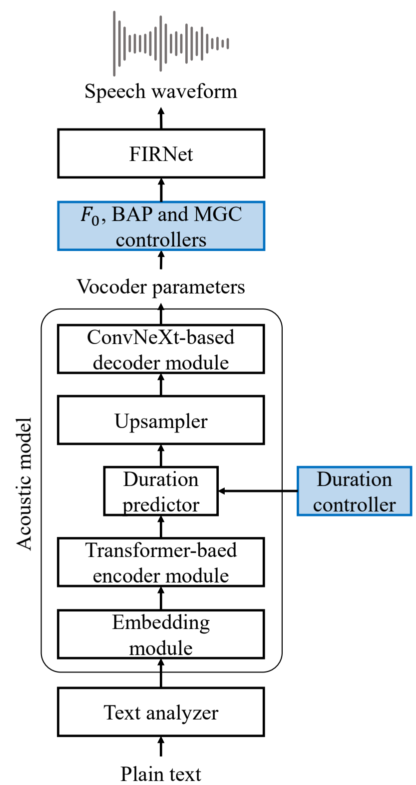

Fig. 5. Neural TTS structure used in the neural TTS application.

These applications highlight FIRNet's value for real-time and flexible voice factor control, with ethical responsibility.

References

[1] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "FastSpeech 2: Fast and high-quality end-to-end text to speech," in *Proc. ICLR*, May 2021.

[2] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," in *Proc. ICML*, 2021, pp. 5530–5540.

[3] D. Lim, S. Jung, and E. Kim, "JETS: Jointly training fastspeech2 and hifi-gan for end to end text to speech," in *Proc. Interspeech 2022*, 2022, pp. 21–25.

[4] T. Okamoto, Y. Ohtani, T. Toda, and H. Kawai, "ConvNeXt-TTS and ConvNeXt-VC: ConvNeXt-based fast end-to-end sequence-to-sequence text-to-speech and voice conversion," in *Proc. ICASSP 2024*, 2024, pp. 12 456–12 460.

[5] T. Okamoto, T. Toda, and H. Kawai, "E2E-S2S-VC: End-To-End Sequence-To-Sequence Voice Conversion," in *Proc. INTERSPEECH*, Aug. 2023, pp. 2043–2047.

[6] K. Tanaka, H. Kameoka, and T. Kaneko, "Prvae-vc: Non-parallel many-to-many voice conversion with perturbation-resistant variational autoencoder," in *SSW2023*, 2023, pp. 88–93.

[7] K. Tanaka, H. Kameoka, T. Kaneko, and Y. Kondo, "Prvae-vc2: Non-parallel voice conversion by distillation of speech representations," in *Proc. INTERSPEECH 2024*, 2024, pp. 4363–4367.

[8] Y. Wang, D. Stanton, Y. Zhang, R. J. Skerry-Ryan, E. Battenberg, J. Shor, Y. Xiao, F. Ren, Y. Jia, and R. A. Saurous, "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis," in *Proc. ICML*, 2018.

[9] E. Cooper, C.-I. Lai, Y. Yasuda, F. Fang, X. Wang, N. Chen, and J. Yamagishi, "Zero-shot multi-speaker text-to-speech with state-of-the-art neural speaker embeddings," in *Proc. ICASSP 2020*, 2020, pp. 6184–6188.

[10] C. Wang, S. Chen, Y. Wu, Z. Zhang, L. Zhou, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li, L. He, S. Zhao, and F. Wei, "Neural codec language models are zero-shot text to speech synthesizers," arXiv, January 2023.

[11] A. Łańcucki, "Fastpitch: Parallel text-to-speech with pitch prediction," in *Proc. ICASSP 2021*, 2021, pp. 6588–6592.

[12] Y. Shirahata, R. Yamamoto, E. Song, R. Terashima, J.-M. Kim, and K. Tachibana, "Period VITS: Variational inference with explicit pitch modeling for end-to-end emotional speech synthesis," in *Proc. ICASSP 2023*, 2023.

[13] J. Lee, W. Jung, H. Cho, J. Kim, and J. Kim, "PITS: Variational pitch inference without fundamental frequency for end-to-end pitch-controllable TTS," in *ICML 2023 Workshop on Structured Probabilistic Inference & Generative Modeling*, 2023.

[14] R. Shimizu, R. Yamamoto, M. Kawamura, Y. Shirahata, H. Doi, T. Komatsu, and K. Tachibana, "PromptTTS++: Controlling speaker identity in prompt-based text-to-speech using natural language descriptions," in *Proc. ICASSP 2024*, 2024, pp. 12 672–12 676.

[15] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "A hidden semi-Markov model-based speech synthesis system," *IEICE Transactions on Information & Systems*, vol. E90-D, no. 5, pp. 825–834, May 2007.

[16] H. Zen, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," in *Proc. ICASSP 2013*, 2013, pp. 7962–7966.

[17] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: A vocoder-based highquality speech synthesis system for real-time applications," *IEICE Trans. Info. & Syst.*, vol. E99-D, no. 7, pp. 1877–1884, Jul. 2016.

[18] T. Chiba and M. Kajiyama, *The Vowel*. Tokyo-Kaiseikan Pub. Co., Ltd., 1942.

[19] Y. Ohtani, T. Okamoto, T. Toda, and H. Kawai, "Fast neural vocoder with fundamental frequency control using finite impulse response filters," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 33, pp. 1893–1906, 2025.

[20] 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," in *MAVEBA 2001*, Sep. 2001.

[21] K. Tokuda, T. Kobayashi, T. Masuko, and S. Imai, "Mel-generalized cepstral analysis," in *Proc. ICASSP*, 1994, pp. 1043–1046.

[22] M. Morise, "CheapTrick, A spectral envelope estimator for high-quality speech synthesis," *Speech Comm.*, vol. 67, pp. 1–7, Mar. 2015.

[23] ——, "D4C, A band-aperiodicity estimator for high-quality speech synthesis," *Speech Comm.*, vol. 84, pp. 57–65, Nov. 2016.

[24] T. Yoshimura, T. Fujimoto, K. Oura, and K. Tokuda, "SPTK4: An open-source software toolkit for speech signal processing," in *12th ISCASpeech Synthesis Workshop (SSW 2023)*, 2023, pp. 211–217.

[25] A. Oppenheim and D. Johnson, "Discrete representation of signals," *Proc. of the IEEE*, vol. 60, no. 6, pp. 681–691, 1972.

[26] T. Okamoto, Y. Shiga, and H. Kawai, "Hi-Fi-CAPTAIN: High-fidelity and high-capacity conversational speech synthesis corpus developed by NICT," https://ast-astrec.nict.go.jp/en/release/hi-fi-captain/, 2023.

[27] J. Ahn, Y. Kim, Y. Choi, D. Kwak, J.-H. Kim, S. Mun, and J. S. Chung, "VoxSim: A perceptual voice similarity dataset," in *Proc. Interspeech*, 2024, pp. 2580–2584.

[28] T. Okamoto, Y. Ohtani, and H. Kawai, "Mobile PresenTra: NICT fast neural text-to-speech system on smartphones with incremental inference of ms-fc-hifi-gan for law-latency synthesis," in *Proc. INTERSPEECH 2024*, 2024, pp. 997–998.

[29] J. Kim, S. Kim, J. Kong, and S. Yoon, "Glow-TTS: A generative flow for text-to-speech via monotonic alignment search," in *Proc. NeurIPS*, 2020, pp. 8067–8077.

[30] S. Woo, S. Debnath, R. Hu, X. Chen, Z. Liu, I.-S. Kweon, and S. Xie, "ConvNeXt V2: Co-designing and scaling convnets with masked autoencoders," *Proc. CVPR*, pp. 16 133–16 142, 2023.