IEICE TRANS. INF. & SYST., VOL.E104–D, NO.7 JULY 2021

PAPER

# Real-Time Full-Band Voice Conversion with Sub-Band Modeling and Data-Driven Phase Estimation of Spectral Differentials

**Takaaki SAEKI**[†a)], ***Nonmember***, **Yuki SAITO**[†b)], **Shinnosuke TAKAMICHI**[†c)], ***and*** **Hiroshi SARUWATARI**[†d)], ***Members***

**SUMMARY** This paper proposes two high-fidelity and computationally efficient neural voice conversion (VC) methods based on a direct waveform modification using spectral differentials. The conventional spectral-differential VC method with a minimum-phase filter achieves high-quality conversion for narrow-band (16 kHz-sampled) VC but requires heavy computational cost in filtering. This is because the minimum phase obtained using a fixed lifter of the Hilbert transform often results in a long-tap filter. Furthermore, when we extend the method to full-band (48 kHz-sampled) VC, the computational cost is heavy due to increased sampling points, and the converted-speech quality degrades due to large fluctuations in the high-frequency band. To construct a short-tap filter, we propose a lifter-training method for data-driven phase reconstruction that trains a lifter of the Hilbert transform by taking into account filter truncation. We also propose a frequency-band-wise modeling method based on sub-band multi-rate signal processing (sub-band modeling method) for full-band VC. It enhances the computational efficiency by reducing sampling points of signals converted with filtering and improves converted-speech quality by modeling only the low-frequency band. We conducted several objective and subjective evaluations to investigate the effectiveness of the proposed methods through implementation of the real-time, online, full-band VC system we developed, which is based on the proposed methods. The results indicate that 1) the proposed lifter-training method for narrow-band VC can shorten the tap length to 1/16 without degrading the converted-speech quality, and 2) the proposed sub-band modeling method for full-band VC can improve the converted-speech quality while reducing the computational cost, and 3) our real-time, online, full-band VC system can convert 48 kHz-sampled speech in real time attaining the converted speech with a 3.6 out of 5.0 mean opinion score of naturalness.

***key words:*** *voice conversion, spectral differentials, deep neural networks, data-driven phase, sub-band modeling*

## 1. Introduction

Voice conversion (VC) converts the characteristics of source speech into those of target speech while keeping the linguistic information unchanged [1]. It has the potential to achieve speech communication beyond the physical constraints of the human vocal organs [2].

The most common VC method is statistical VC [3], [4], which is used to construct an acoustic model that converts speech features of a source speaker into those of a target speaker. Deep neural network (DNN)-based VC [5], [6] has been widely studied, and many models for achieving higher-converted-speech quality have been proposed. From a practical point of view, VC must be real-time and online with limited computational resources, and real-time VC methods based on a Gaussian mixture model [7] and DNN [8] have been studied. They achieve online conversion of narrow-band (16 kHz-sampled) speech using a single CPU on a laptop PC. However, their computational cost is still high, and we need to reduce this cost towards portable (e.g., VC using a low-power CPU on a smart phone) or full-band (48 kHz-sampled) VC that covers the human audible range.

VC consists of three steps: feature analysis, feature conversion, and waveform synthesis. For the last step, which is the most computationally exhaustive part, we focus on a spectral-differential VC method [9] that performs conversion in the waveform-domain by applying a spectral differential filter to the source speech waveform. This 1) achieves high-quality conversion by avoiding vocoder errors and 2) incurs less computational cost than neural vocoders [10]–[12] that use large DNNs and require sample-by-sample heavy computation. Spectral-differential VC method originally used a mel-log spectrum approximation (MLSA) filter [13] to filter a source speech, but Suda et al. found that using a minimum-phase filter achieved higher converted-speech quality than using the MLSA filter [14]. Regarding the minimum-phase filter, an acoustic model (e.g., DNN) outputs a real cepstrum of the converted speech, and the Hilbert transform using a lifter with fixed parameters determines the phases of the filter from the real cepstrum. These processes are suitable for our aim because their computational costs (i.e., filter design) are very low. However, since the minimum-phase filter is not guaranteed to have a short tap length (i.e., the number of samples of the filter), it increases the computational cost of filtering. Furthermore, there are two problems when we extend this method from narrow-band VC to full-band VC: 1) converted-speech quality degrades due to large fluctuations in the high-frequency band, and 2) computational cost is high (mainly in the filtering operation) due to increased sampling points.

We propose two methods to achieve real-time and high-fidelity conversion. First, we propose a lifter-training method with filter truncation for significantly reducing computational cost without degrading converted-speech quality. This method jointly trains not only a DNN-based acoustic model but also a lifter with trainable parameters. Since parameters of the DNNs and the lifter are optimized to maxi-

Manuscript received December 2, 2020.
Manuscript revised February 28, 2021.
Manuscript publicized April 16, 2021.

[†]The authors are with the University of Tokyo, Tokyo, 113–8656 Japan.
a) E-mail: takaaki_saeki@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: hiroshi_saruwatari@ipc.i.u-tokyo.ac.jp

DOI: 10.1587/transinf.2020EDP7252

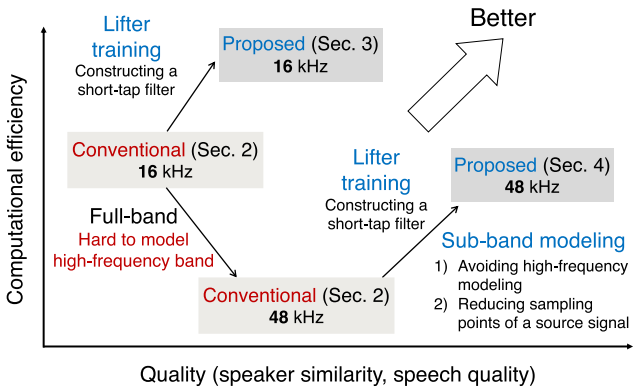


**Fig. 1** Overview of conventional method, proposed lifter training method and proposed sub-band modeling method. In Sect. 5, we present implementation of our real-time, online, full-band VC system based on proposed methods.

mize conversion accuracy with the consideration of a truncated (i.e., short-tap) filter, this method can reduce the computational cost while preserving conversion accuracy. The main difference between our method and the conventional spectral-differential VC method using a minimum-phase filter is with the lifter to determine the phase of the filter. Whereas the lifter of the minimum-phase filter is *fixed*, that of our method is *trained* from speech data to determine the phases of a truncated filter. Our lifter-training method can be viewed as a framework of DNN-based phase reconstruction from the amplitude spectrum [15]. Second, for full-band VC, we also propose a frequency-band-wise modeling method based on sub-band multi-rate signal processing (hereafter, "sub-band modeling method") [16]. Since the characteristics of a speech waveform vary significantly from band to band, it is effective to process the waveform separately for each band. In sub-band WaveNet [17], the speech waveform is divided into several bands and downsampled, and the waveform in each band is processed separately. This method enhances the computational efficiency by reducing sampling points of signals converted with filtering and improves the converted-speech quality by modeling only the low-frequency band that contributes to speaker identity and avoiding high-frequency modeling. Figure 1 shows an overview of our proposed methods. We apply our lifter-training method to narrow-band VC to significantly reduce computational cost and achieve real-time VC with a low-power CPU of a single-board computer (e.g., Raspberry Pi). Furthermore, our sub-band modeling method for full-band VC achieves real-time conversion with a single CPU of a mobile device. We also present implementation of the real-time online VC systems based on our proposed methods. This system is highly applicable because it supports F0 transformation and online conversion. Experimental results indicate that 1) the proposed lifter-training method for narrow-band VC can shorten the tap length to 1/16 without degrading converted-speech quality and 2) the proposed sub-band modeling method for full-band VC can improve the converted-speech quality while reducing computational cost, and 3) our online VC system can convert 48 kHz-sampled speech in real time attaining converted speech with a 3.6 out of 5.0 mean opinion score (MOS) of naturalness.

In Sect. 2, we describe the conventional spectral-differential VC method with a minimum-phase filter. We describe data-driven phase reconstruction with our lifter-training method for short-tap filtering in Sect. 3 and our sub-band modeling method for full-band VC in Sect. 4. In Sect. 5, we present the implementation of our online full-band VC system. We explain the objective and subjective evaluations and the results in Sect. 6 and conclude this paper in Sect. 7. The main contributions of this work are as follows:

- We propose a liftering-based phase-estimation method with filter truncation. This method reduces the computational cost for filtering without lowering conversion accuracy. This is also presented in our conference paper [18].
- We propose a sub-band modeling method for full-band VC. It improves full-band converted-speech quality and provides new insights into high-frequency processing of a speech signal that can be applied to various tasks.
- We implement the real-time, online, full-band VC system based on the proposed methods. We presented an overview and demonstration of this system in our demo paper [19]. In this paper, we describe the structure and evaluation results of our system in detail. Furthermore, we introduce several enhancement techniques for a higher-quality real-time VC system. These techniques include our proposed F0 equalization method, which can be applied to other VC frameworks to improve feature analysis.

## 2. Spectral-Differential VC with Minimum-Phase Filter

This section describes the training and conversion processes of the conventional spectral-differential VC method with a minimum-phase filter (hereafter, "conventional method").

### 2.1 Training Process

Let $\boldsymbol{F}^{(\mathrm{X})} = \left[\boldsymbol{F}_1^{(\mathrm{X})\top}, \ldots, \boldsymbol{F}_t^{(\mathrm{X})\top}, \ldots, \boldsymbol{F}_T^{(\mathrm{X})\top}\right]^\top$ be a complex frequency spectrum sequence obtained by applying the short-time Fourier transform (STFT) to an input speech waveform, where $t$ represents the frame index and $T$ is the total number of frames. For simplicity, we focus on frame $t$. A low-order real cepstrum $\boldsymbol{C}_t^{(\mathrm{X})}$ can be extracted from $\boldsymbol{F}_t^{(\mathrm{X})}$ [20]. The DNNs then estimate a real cepstrum of differential filter $\hat{\boldsymbol{C}}_t^{(\mathrm{D})}$ from $\boldsymbol{C}_t^{(\mathrm{X})}$. The loss function for $t$ is calculated as $L_t^{(\mathrm{MSE})} = \left(\boldsymbol{C}_t^{(\mathrm{Y})} - \hat{\boldsymbol{C}}_t^{(\mathrm{Y})}\right)^\top \left(\boldsymbol{C}_t^{(\mathrm{Y})} - \hat{\boldsymbol{C}}_t^{(\mathrm{Y})}\right)$, where $\hat{\boldsymbol{C}}_t^{(\mathrm{Y})}$ is a real cepstrum of converted speech given as $\hat{\boldsymbol{C}}_t^{(\mathrm{Y})} = \boldsymbol{C}_t^{(\mathrm{X})} + \hat{\boldsymbol{C}}_t^{(\mathrm{D})}$, and $\boldsymbol{C}_t^{(\mathrm{Y})}$ is a real cepstrum of the target speech.

The DNNs are trained to minimize the loss function for all time frames represented as follows:

$$L^{(\mathrm{MSE})} = \frac{1}{T}\sum_{t=1}^{T} L_t^{(\mathrm{MSE})}. \tag{1}$$

### 2.2 Conversion Process

The $\hat{\boldsymbol{C}}_t^{(\mathrm{D})}$ is estimated with the DNNs. After the high-order components of the cepstrum are padded with zeros, $\hat{\boldsymbol{C}}_t^{(\mathrm{D})}$ is multiplied by a time-independent lifter $\boldsymbol{u}_{\min}$ for a minimum-phase filter. The complex frequency spectrum of differential filter $\hat{\boldsymbol{F}}_t^{(\mathrm{D})}$ can be obtained by taking the inverse discrete Fourier transform (IDFT) of the liftered cepstrum. The lifter $\boldsymbol{u}_{\min}$ is represented as follows [21]:

$$\boldsymbol{u}_{\min}(n) = \begin{cases} 1 & (n = 0, n = N/2) \\ 2 & (0 < n < N/2), \\ 0 & (n > N/2) \end{cases} \tag{2}$$

where $N$ is the number of frequency bins of the DFT. A differential filter in the time domain $\hat{\boldsymbol{f}}_t^{(\mathrm{D})}$ is obtained by applying the IDFT to $\hat{\boldsymbol{F}}_t^{(\mathrm{D})}$. The tap length of $\hat{\boldsymbol{f}}_t^{(\mathrm{D})}$ is equal to $N$.

### 2.3 Trade-off between Computational Cost and Converted-Speech Quality

The most computationally expensive step of the conversion process described in Sect. 2.2 is that of convolving the differential filter into the source speech waveform. To reduce computational cost, we can introduce a simple method of truncating the differential filter $\hat{\boldsymbol{f}}_t^{(\mathrm{D})}$ with a fixed tap length $l$ ($l < N$). For example, when the filter length $N = 512$, we can reduce the computational cost of filtering by 1/4 by setting $l = 128$ and performing the convolution using only the first 128 samples of the 512-tap filter. We define the $l$-tap truncated filter as $\hat{\boldsymbol{f}}_t^{(l)}$. Since the power of the minimum-phase filter is concentrated around 0, it is possible to truncate up to a certain length without losing the converted-speech quality. When we increase $l$, converted-speech quality does not degrade, but the computational cost of the filtering operation increases. On the other hand, when we decrease $l$, we can efficiently reduce computational cost, but $\hat{\boldsymbol{f}}_t^{(l)}$ degrades converted-speech quality.

### 2.4 Extension to Full-Band VC

When we apply the conventional method to full-band VC, there are two problems, i.e., 1) converted-speech quality degrades due to large fluctuations in the high-frequency band, and 2) computational cost is high (mainly in the filtering operation) due to increased sampling points. Problem 1 is that the high-frequency components with high variability are difficult to predict using a statistical model due to the low correlation between speakers. Problem 2 occurs because the computational cost of the filtering operation depends on the signal length and filter length, and both lengths increase as the sampling frequency increases.

## 3. Data-Driven Phase Reconstruction with Lifter Training

In this section, we present the training and conversion processes of our lifter-training method. The main difference between this method and the conventional one is with the lifter to determine the phase of the filter, as shown in Fig. 2.

### 3.1 Training Process

Our lifter-training method trains not only DNNs but also a lifter to avoid converted-speech-quality degradation caused by filter truncation. Let $\boldsymbol{u} = [u_1, \ldots, u_c]^\top$ be a time-independent trainable lifter, where $c$ is the dimension of the real cepstrum. The filter-truncation process with $l$ is integrated into the training, as shown in Fig. 3.

As we described in Sect. 2.1, the DNNs estimate $\hat{\boldsymbol{C}}_t^{(\mathrm{D})}$ from $\boldsymbol{C}_t^{(\mathrm{X})}$. Then $\hat{\boldsymbol{C}}_t^{(\mathrm{D})}$ is multiplied by the trainable lifter $\boldsymbol{u}$, and the complex frequency spectrum of the differential filter $\hat{\boldsymbol{F}}_t^{(\mathrm{D})}$ is obtained from the IDFT of $\hat{\boldsymbol{C}}_t^{(\mathrm{D})}$ and exponential calculation. The differential filter in the time domain $\hat{\boldsymbol{f}}_t^{(\mathrm{D})}$ is obtained by applying the IDFT to $\hat{\boldsymbol{F}}_t^{(\mathrm{D})}$. The $\hat{\boldsymbol{f}}_t^{(\mathrm{D})}$ is truncated

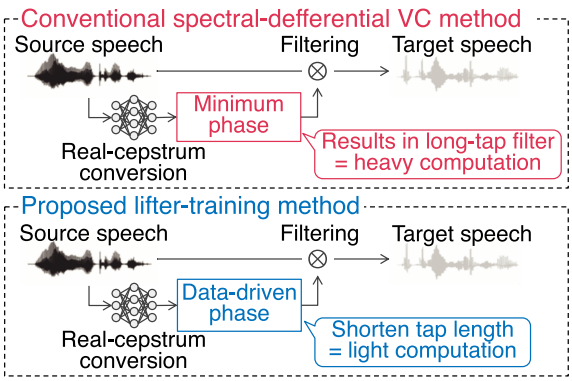


**Fig. 2** Comparison of proposed lifter-training method and conventional method.

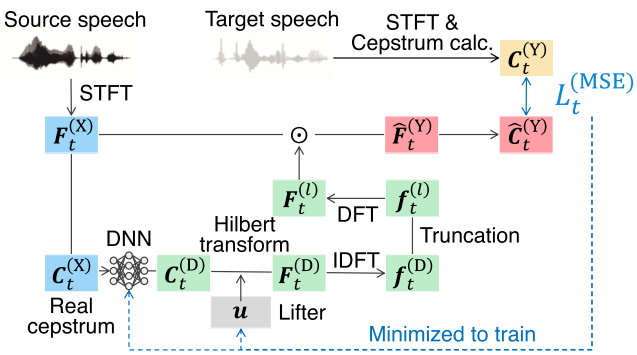


**Fig. 3** Procedure of proposed lifter-training method.

to $\hat{\boldsymbol{f}}_t^{(l)}$ by applying a window function $\boldsymbol{w}$ given as:

$$\hat{\boldsymbol{f}}_t^{(l)} = \hat{\boldsymbol{f}}_t^{(\mathrm{D})} \cdot \boldsymbol{w}, \tag{3}$$

$$\boldsymbol{w} = \left[\overbrace{1, \cdots,}^{0\text{th}} \overbrace{1}^{(l-1)\text{th}}, \overbrace{0, \cdots,}^{l\text{th}} \overbrace{0}^{(N-1)\text{th}}\right]^{\top}. \tag{4}$$

By using the DFT again, a complex spectrum of the $l$-tap truncated differential filter $\hat{\boldsymbol{F}}_t^{(l)}$ can be obtained. A complex spectrum of converted speech $\hat{\boldsymbol{F}}_t^{(\mathrm{Y})}$ is obtained by multiplying $\boldsymbol{F}_t^{(\mathrm{X})}$ by $\hat{\boldsymbol{F}}_t^{(l)}$, and the real cepstrum of converted speech $\hat{\boldsymbol{C}}_t^{(\mathrm{Y})}$ is extracted from $\hat{\boldsymbol{F}}_t^{(\mathrm{Y})}$. The parameters of the DNNs and the lifter are jointly trained to minimize the same loss function as Eq. (1). Since all processes of this method are differentiable, the training can be done by back-propagation [22].

### 3.2 Conversion Process

In the conversion process, the trained DNNs and lifter estimate $\hat{\boldsymbol{F}}_t^{(\mathrm{D})}$. The $\hat{\boldsymbol{f}}_t^{(\mathrm{D})}$ is obtained by applying the IDFT to $\hat{\boldsymbol{F}}_t^{(\mathrm{D})}$, and $\hat{\boldsymbol{f}}_t^{(l)}$ is obtained by truncating with $l$. We can obtain the converted speech waveform by applying $\hat{\boldsymbol{f}}_t^{(l)}$ to the source speech waveform.

### 3.3 Discussion

With the conventional method, the cepstrum is multiplied by the lifter coefficient to determine the shape of the filter to have minimum phase. Although the shape of the differential filter changes due to truncation, it is transformed to compensate for the effect of the truncation by applying the Hilbert transform using the lifter trained with the proposed lifter-training method. As a result, our lifter-training method can reduce the calculation amount while suppressing converted-speech quality degradation caused by the filter truncation. Figure 4 shows the cumulative power distribution of the differential filter with the conventional method ($l = 512$) and proposed lifter-training method ($l = 32$). The values on the vertical axis are normalized with the cumulative total. We can see that the proposed lifter-training method concentrates the power in the short taps whereas the conventional method does not. Figure 5 also shows the difference between the lifter trained with the proposed method ($l = 64$) and that for minimum phasing. The trained lifter is entirely different from that with the conventional method and has a complicated shape. Figure 6 shows zero plots with truncated ($l = 32$) differential filters using the conventional method and the proposed lifter-training method. Some zeros are distributed outside the unit circle in the conventional method because the shape of the filter changes by truncating the estimated minimum-phase filter. The proposed lifter-training method works to correct the distribution of the zeros to the inside of the unit circle, suggesting that the proposed lifter-training method compensates for the shape change of the filter caused by filter truncation and estimate short-tap

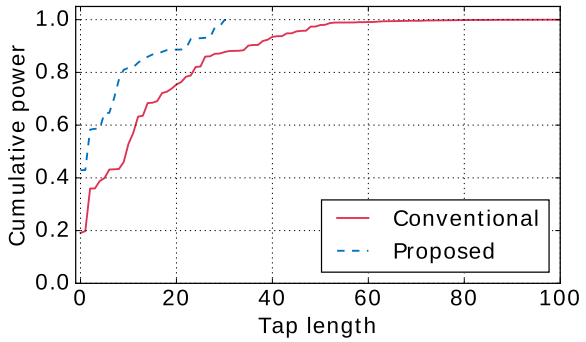


**Fig. 4** Cumulative power distributions of differential filter with conventional method and proposed lifter-trainig method.

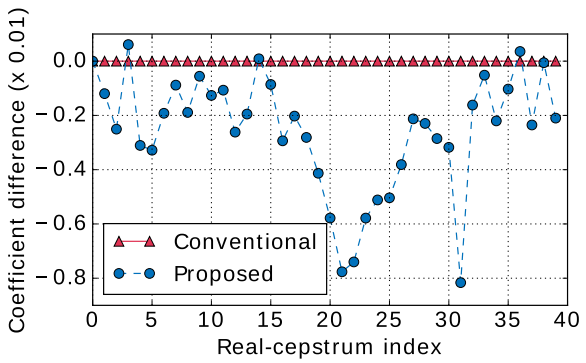


**Fig. 5** Difference between lifter trained with proposed lifter-trained method ($l = 64$) and that for minimum phasing with conventional method.

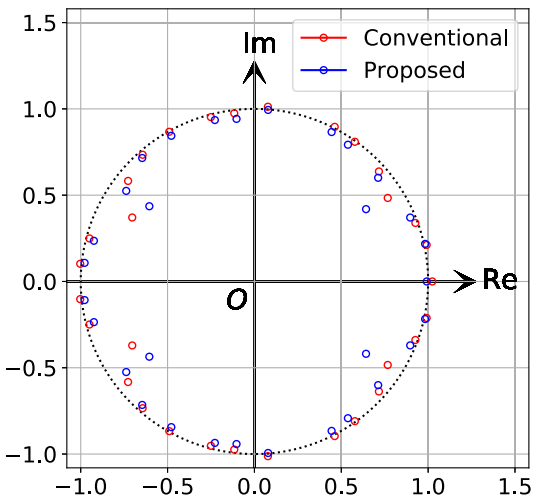


**Fig. 6** Zero plots of differential filters with conventional method and proposed lifter-training method.

filter while avoiding accuracy deterioration. Furthermore, most of the zeros with the conventional method are located near the unit circle, while the zeros with the proposed lifter-training method are relatively far from the circle. This result indicates that the proposed lifter-training method flattens the amplitude-frequency characteristics of the differential filter. Note that we used the female-to-female data pairs described

in Sect. 6.1 and down-sampled them to 16 kHz to get the results shown in Fig. 4 and Fig. 5.

As explained in Sect. 1, liftering-based phase estimation requires only small computation. Since our lifter-training method adopts the same estimation as the conventional method, there is no increase in computational cost of phase estimation.

We applied our lifter-training method to VC, i.e., speaker conversion. We expect that this method can be applied to other tasks processed by filtering, e.g., source separation and speech enhancement.

## 4. Frequency-Band-Wise Modeling with Sub-Band Multirate Processing

As described in Sect. 2.4, when we use the conventional method for full-band VC, 1) converted-speech quality degrades due to large fluctuations in the high-frequency band, and 2) computational cost is high (mainly in the filtering operation) due to increased sampling points. We use our sub-band modeling method to solve these problems. This method divides the full-band source speech into multiple sub-band signals and only converts the lowest-band signal with the differential filter. Figure 7 shows the workflow of this method. After the full-band signal is divided into sub-band signals by sub-band analysis (Sect. 4.1), they are converted with the trained model (Sect. 4.2), and the full-band converted speech is obtained by sub-band synthesis (Sect. 4.3).

The 0–8 kHz signal converted with this method is consistent with the bandwidth handled with the conventional method for narrow-band VC, and with the bandwidth of wide-band speaker verification [23]. Therefore, it is reasonable to focus on this bandwidth in converting speaker identity. Since 8–24 kHz signal contributes to speech quality, we can enhance the output-speech quality by directly using the input signal. Unlike other VC methods, such as seq-to-seq VC [24]–[26], the number of frames of the lowest-band signal does not change between the input and output speech.

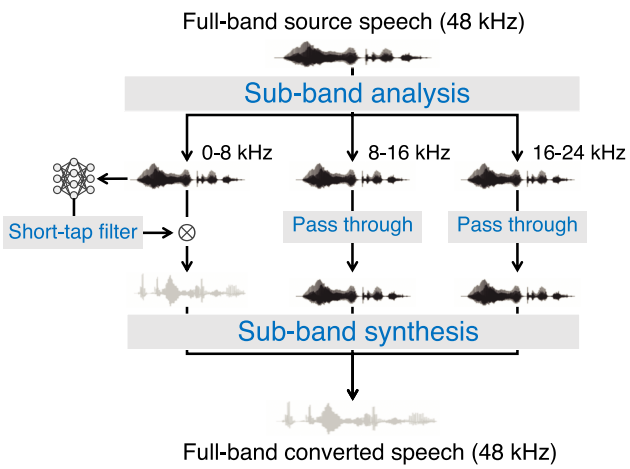


**Fig. 7** Workflow of our sub-band modeling method for full-band VC.

Since the converted-lowest-band signal is frame-wise synchronized with the higher-band signals, we can directly synthesize the full-band converted speech without time alignment.

### 4.1 Sub-Band Analysis

An original full-band signal $x(t)$ is divided into $N$ sub-band streams ($N = 3$ in this paper), and modulated by $W_N^{-t(n-1/2)}$ and shifted to the base band (Fig. 8 (a)):

$$x_n(t) = x(t)\, W_N^{-t(n-1/2)}, \tag{5}$$

where $n = 1, 2, \cdots, N$ is a frequency-band index, and $W_N = \exp(j2\pi/2N)$. Then $x_n(t)$ is bandlimited using low-pass filter $f(t)$ (Fig. 8 (b)):

$$x_{n,pp}(t) = f(t) * x_n(t), \tag{6}$$

where the cutoff frequency of $f(t)$ is $\pi/2N$, and $*$ represents the convolution operator. By introducing single-sideband (SSB) modulation, real-valued signal $x_{n,\mathrm{SSB}}(t)$ is obtained (Fig. 8 (c)):

$$x_{n,\mathrm{SSB}}(t) = x_{n,pp}(t)\, W_N^{t/2} + x_{n,pp}^*(t)\, W_N^{-t/2}, \tag{7}$$

where $\cdot^*$ denotes the complex conjugate. The $n$-th sub-band waveform $x_n(k)$ is obtained with decimation (Fig. 8 (d)):

$$x_n(k) = x_{n,\mathrm{SSB}}(kM). \tag{8}$$

### 4.2 Training and Conversion Processes

In the training process, we train the model as described in Sect. 2.1 or Sect. 3.1 using only the lowest-band signal ($n = 1$). This training process improves the converted-speech quality by modeling only the low-frequency band that contributes to speaker identity and avoiding high-frequency modeling. In the conversion process, only the lowest-band

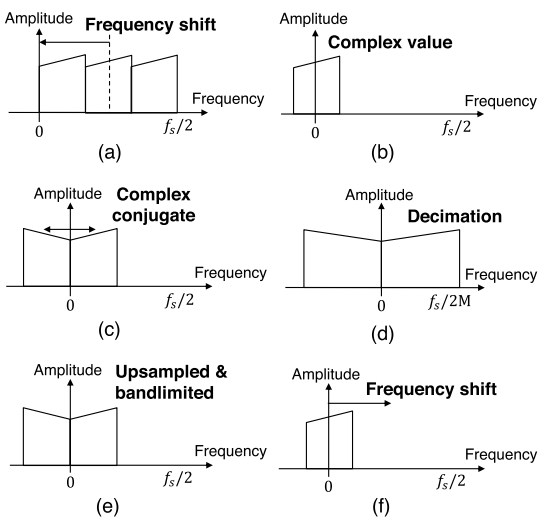


**Fig. 8** Procedures of sub-band analysis and synthesis.

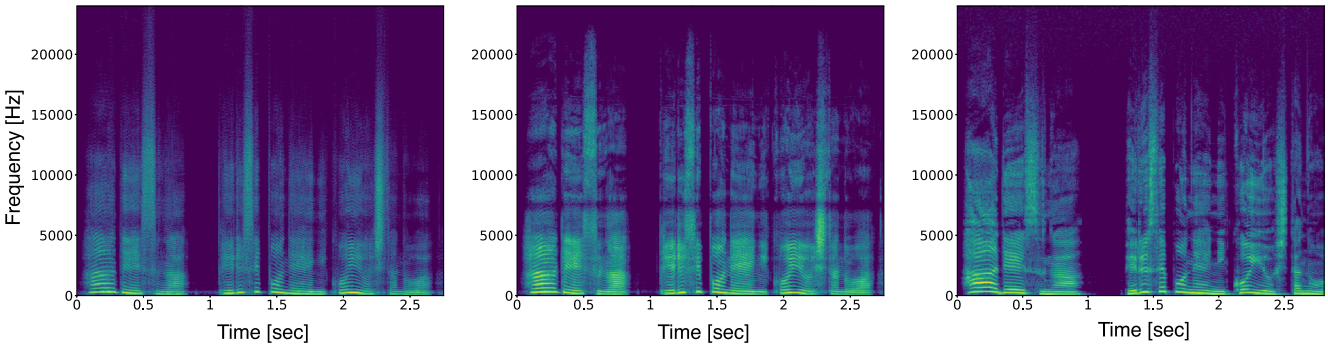


**Fig. 9** Spectrograms of (a) converted speech obtained by applying differential filter to full-band source speech, (b) converted speech obtained by applying differential filter to only lowest-band signal, and (c) full-band target speech.

signal is converted, as described in Sect. 2.2 or Sect. 3.2, and higher-band signals are not converted. We can enhance computational efficiency by using this conversion because it reduces sampling points of signals converted with filtering.

### 4.3 Sub-Band Synthesis

To synthesize a full-band signal, the converted sub-band signals $\hat{x}_n(t)$ are up-sampled as follows:

$$\hat{x}_{n,\mathrm{SSB}}(t) = \begin{cases} \hat{x}_n(t/M) & (t = 0, M, 2M, \cdots) \\ 0 & (\text{otherwise}). \end{cases} \tag{9}$$

The $\hat{x}_{n,\mathrm{SSB}}(t)$ is shifted to the base band, and bandlimited with low-pass filter $g(t)$ (Fig. 8 (e)):

$$\hat{x}_{n,pp}(t) = g(t) * \left(\hat{x}_{n,\mathrm{SSB}}(t)\, W_N^{-t/2}\right). \tag{10}$$

Finally, the full-band signal $\hat{x}(t)$ is synthesized (Fig. 8 (f)):

$$\hat{x}(t) = \sum_{n=1}^{N} \left\{\hat{x}_{n,pp}(t)\, W_N^{t(n-1/2)} + \hat{x}^*_{n,pp}(t)\, W_N^{-t(n-1/2)}\right\}. \tag{11}$$

### 4.4 Discussion

The number of sub-band streams $N$ is a hyperparameter. When we increase $N$, the bandwidth to pass through the input signal increases. This enhances speech quality but degrades speaker similarity. On the other hand, when we decrease $N$, speech quality and computational efficiency decrease because the bandwidth to convert the input signal increases. As a result of a preliminary experiment, we use $N = 3$ as shown in Fig. 7, which achieves the best speaker similarity and speech quality.

In this study, we passed through the mid-band (8–16 kHz) and high-band (16–24 kHz) signals. The simplest way to further improve speaker similarity is to convert the mid-band and high-band signals by using statistical models. In a preliminary experiment, we evaluated the method of converting the mid-band and high-band signals by using a DNN and confirmed that the converted-speech quality degraded.

Figure 9 shows the spectrograms of the converted speech obtained by applying the differential filter to the full-band source speech (defined as "benchmark" in Sect. 6), the converted speech obtained by applying the filter to only the lowest-band signal, and the full-band target speech. In these results, we used the female-to-female data pairs described in Sect. 6.1. When we apply the differential filter to the full-band source speech, the accuracy of estimating the differential spectrum by using a DNN degrades and we can observe the over-smoothing of the spectrum in the whole band (Fig. 9 (a)). When we apply the differential filter only to the lowest band, however, the DNN can estimate the differential spectrum of the lowest band with high accuracy, and we can observe the fine structures of the spectrum (Fig. 9 (b)).

Our sub-band modeling method can significantly reduce the computational cost for full-band VC because it can decrease both the source-signal length and the filter length. Furthermore, we can use our lifter-training method with filter truncation when we convert the lowest-band signal and can further reduce the computational cost of the filtering operation.

## 5. Implementation of Real-Time, Online, Full-Band VC System

In Sects. 3 and 4, we presented computationally efficient and high-fidelity full-band VC methods respectively. We now present the implementation of our online full-band VC system by combining these methods. Figure 10 shows the pipeline of our system. It receives a 5-ms waveform of source speech and outputs a 5-ms waveform of the converted speech. In this section, we also present several methods for enhancing the performance of our online VC system without increasing the computational cost during conversion.

### 5.1 Basic Structure

We describe the basic structure of our online full-band VC

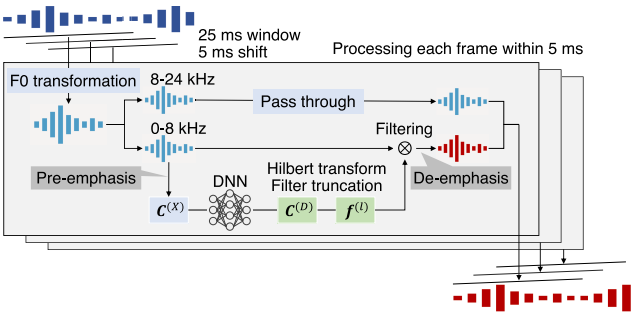


**Fig. 10** Pipeline of our real-time, online, full-band VC system.

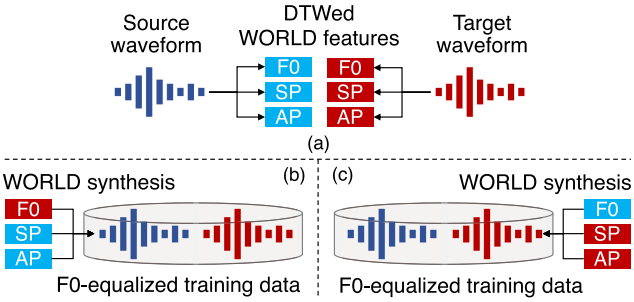


**Fig. 11** Procedure of our F0 equalization methods in pre-processing. (a) We first obtain DTWed WORLD features. "SP" and "AP" indicate spectral envelope and aperiodicity, respectively. Then we have two options for equalizing F0: (b) F0 of source speech is replaced with that of target speech and (c) its inverse procedure. Re-synthesized waveform becomes a new source or target speech waveform of training data. When using F0 transformation described in Sect. 5.1.2, we apply it to source speech in advance.

system, which consists of analysis, conversion, and synthesis steps.

#### 5.1.1 Analysis Step

In the analysis step, our system extracts the input feature of the DNN. First, we apply the Hanning window to the input frame obtained from full-band source speech and use the sub-band multi-rate signal processing described in Sect. 4. To reduce the redundancy of the source cepstrum extracted from the 0–8 kHz signal, we apply a first-order pre-emphasis filter $E(z) = 1 - \alpha z^{-1}$ to the lowest-band signal, with $\alpha = 0.97$. In our preliminary experiments, we found that this pre-emphasis processing improved converted-speech quality of the system. We then extract the low-order cepstrum $\boldsymbol{C}^{(X)}$ by applying DFT analysis to the frame of the lowest-band signal.

#### 5.1.2 Conversion Step

In the conversion step, our system constructs a time-domain differential filter from $\boldsymbol{C}^{(X)}$, as mentioned in Sect. 3. The DNN estimates the real cepstrum of the differential filter $\hat{\boldsymbol{C}}^{(\mathrm{D})}$ from the real cepstrum of the source speech $\boldsymbol{C}^{(X)}$, and we construct the truncated differential filter $\hat{\boldsymbol{f}}^{(l)}$ from the real cepstrum using a minimum-phase filter or data-driven phase proposed in Sect. 3.

Since spectral-differential VC method can only convert vocal tract characteristics, we incorporate F0 transformation into our system for cross-gender conversion using a direct waveform modification with PICOLA [27]. This method is more computationally efficient and suitable for our purpose than vocoder-based F0 transformation.

#### 5.1.3 Synthesis Step

In the synthesis step, we obtain the converted speech by applying the truncated differential filter $\hat{\boldsymbol{f}}^{(l)}$ to the source speech waveform. Then we apply the de-emphasis filter $D(z) = 1/\left(1 - \alpha z^{-1}\right)$ to the converted-lowest-band signal. We do not convert the higher-band signals and pass through them. We can obtain the frame of the full-band converted signal by combining the processed lowest-band signal and higher-band signals. Finally, we overlap-add the frame to the previous calculation results and the first 5-ms waveform is output.

### 5.2 Methods for Enhancing Performance of our Online VC System

We present several methods for enhancing naturalness and speaker similarity of converted speech obtained with our online VC system. Since all the methods are for training data refinement or DNN training, they do not increase the computational cost of our system during conversion.

#### 5.2.1 F0 Equalization in Pre-Processing

In the analysis step of our VC system, we should calculate the spectral envelope component independently of the excitation components. The well-known method for estimating the spectral envelope is a high-quality vocoder, e.g., WORLD [28]. However, it is not practical in real-time VC due to its high computational cost and large time delays for analysis. We use a real cepstrum of a DFT spectrum[†]. However, a real cepstrum of a DFT spectrum suffers from the excitation component [29]. This fact affects not only the analysis step but also the conversion step; the DNN has to predict the excitation differences between speakers in addition to spectral-envelope differences. Such prediction becomes more difficult than the prediction of only spectral-envelope differences and degrades the prediction accuracy. Therefore, we use data refinement methods so that the DNN predicts only spectral-envelope differences.

Figure 11 shows these methods. The essential point is to remove F0 differences between speakers, i.e., we equalize one speaker's F0 to the other speaker's one. After aligning

[†]The most simple solution is to use the vocoder during only training. In this solution, we use a real cepstrum of the WORLD's spectral envelope during training and use that of a DFT spectrum during conversion. However, in our preliminary experiment, we found that such a method significantly degraded converted-speech quality.

the source speaker's frames and target speaker's frames using the dynamic time warping (DTW) algorithm, we obtain temporally aligned F0, a spectral envelope, and aperiodicity using WORLD (Fig. 11 (a)). We have two options to equalize the F0s; equalizing the source speaker's F0 to the target speaker's (Fig. 11 (b)) or its inverse procedure (Fig. 11 (c)). The former replaces F0 of the source speech with that of the target speech and synthesizes a speech waveform. The synthesized waveform is used as a new source speech waveform of the training data. The latter is their inverse, i.e., a method that exchanges "source" and "target" of the above sentences. When using a real-time F0 transformation method (see 2nd paragraph of Sect. 5.1.2) during conversion, we apply this method to the source speech and carry out the above F0 equalization.

The above pre-processing of the training data efficiently removes F0 differences between speakers. Therefore, prediction by using a DNN is expected to become less affected by F0.

#### 5.2.2 Vocoder-Guided Training

The F0 equalization method uses a vocoder to alleviate the effect of F0 differences in the training data. In this section, we present a method of using a vocoder for DNN training to enhance the alleviation effect. As a pre-process, we extract the spectral envelopes of the source speech and target speech with WORLD because it is more robust against F0 compared with DFT-based analysis. From the source and target speech in the training data, we extract not only real cepstra of DFT spectra, $\boldsymbol{C}_t^{(X)}$ and $\boldsymbol{C}_t^{(Y)}$, but also those of WORLD spectral envelopes denoted as $\boldsymbol{c}_t^{(X)}$ and $\boldsymbol{c}_t^{(Y)}$. In DNN training, we add the extra term $L^{(\mathrm{VOC})}$ as

$$L^{(\mathrm{MSE})} + \lambda L^{(\mathrm{VOC})} = \frac{1}{T}\sum_{t=1}^{T}\left(\boldsymbol{C}_t^{(\mathrm{Y})} - \hat{\boldsymbol{C}}_t^{(\mathrm{Y})}\right)^{\top}\left(\boldsymbol{C}_t^{(\mathrm{Y})} - \hat{\boldsymbol{C}}_t^{(\mathrm{Y})}\right) + \frac{\lambda}{T}\sum_{t=1}^{T}\left(\boldsymbol{c}_t^{(\mathrm{D})} - \hat{\boldsymbol{C}}_t^{(\mathrm{D})}\right)^{\top}\left(\boldsymbol{c}_t^{(\mathrm{D})} - \hat{\boldsymbol{C}}_t^{(\mathrm{D})}\right), \tag{12}$$

where $\lambda$ is a weight parameter of vocoder-guided training and $\boldsymbol{c}_t^{(\mathrm{D})} = \boldsymbol{c}_t^{(Y)} - \boldsymbol{c}_t^{(X)}$. This training method works to match the predicted spectral differentials of the DFT spectra and those of the WORLD spectral envelopes. Since $\boldsymbol{c}_t^{(\mathrm{D})}$ is ideally independent on F0, this training helps predict F0-independent spectral differentials. Note that, we cannot add a loss function that directly matches $\boldsymbol{c}_t^{(Y)}$ and $\hat{\boldsymbol{C}}_t^{(Y)}$. This is because $\hat{\boldsymbol{C}}_t^{(Y)}$ is explicitly calculated by DFT and IDFT.

#### 5.2.3 Statistical Compensation Training

The well-known method for improving VC quality is to compensate for the statistics of the converted features, e.g., GAN-based compensation [30]. We now introduce global variance (GV) compensation [4], which alleviates the over-smoothing effect of converted spectra. We can write the full objective by adding the loss term for the GV compensation as

$$\begin{aligned} &L^{(\mathrm{MSE})} + \mu L^{(\mathrm{GV})} \\ &= \frac{1}{T}\sum_{t=1}^{T}\left(\boldsymbol{C}_t^{(\mathrm{Y})} - \hat{\boldsymbol{C}}_t^{(\mathrm{Y})}\right)^{\top}\left(\boldsymbol{C}_t^{(\mathrm{Y})} - \hat{\boldsymbol{C}}_t^{(\mathrm{Y})}\right) \\ &+ \frac{\mu}{T}\sum_{t=1}^{T}\left\{\left(\boldsymbol{C}_t^{(\mathrm{Y})} - \frac{1}{T}\sum_{\tau=1}^{T}\boldsymbol{C}_\tau^{(\mathrm{Y})}\right)^2 - \left(\hat{\boldsymbol{C}}_t^{(\mathrm{Y})} - \frac{1}{T}\sum_{\tau=1}^{T}\hat{\boldsymbol{C}}_\tau^{(\mathrm{Y})}\right)^2\right\}. \end{aligned} \tag{13}$$

## 6. Evaluations

We first investigated the effectiveness of our proposed methods: lifter training described in Sect. 3 and sub-band modeling described in Sect. 4. In this evaluation, we implemented the proposed methods and conventional method in the form of offline conversion and created two intra-gender VC cases: for female-to-female (f2f) and male-to-male (m2m) conversion. We also evaluated the computational efficiency and converted-speech quality of our online VC systems based on the proposed methods and incorporating several improvements. Note that we implemented the online narrow-band VC system in the same manner as Sect. 5. In addition to the intra-gender VC cases, we also evaluated two cross-gender VC cases: female-to-male (f2m) and male-to-female (m2f) for this evaluation.

### 6.1 Evaluation Conditions

The source and target speakers in the f2f case were stored in the JSUT corpus [31] and Voice Actress Corpus [32], respectively. Those in m2m, f2m and m2f cases were stored in the JVS corpus [31]. We used 100 utterances (approx. 12 min.) of each speaker, and the numbers of utterances for training, validation, and test data were 80, 10, and 10, respectively.

When analyzing the narrow-band (16 kHz) signal, the window length was 25 ms, frame shift was 5 ms, the fast Fourier transform (FFT) length was 512 samples, and number of dimensions of the cepstrum was 40 (0th-through-39th). When applying the conventional method to full-band (48 kHz) VC, as described in Sect. 2.4, the window length and frame shift were the same as those in the narrow-band case, but the FFT length was 2048 samples, and number of dimensions of the cepstrum was 120 (0th-through-119th). For pre-processing, the silent intervals of training and validation data were removed, and the lengths of the source and target speech were aligned using DTW.

The DNN architecture of the acoustic model was multilayer perceptron consisting of two hidden layers. We determined the hyperparameters of the DNN using Optuna [33], with the numbers of each hidden unit set to 280 and 100 for the narrow-band signal and set to 840 and 300 when applying the conventional method to full-band VC without our sub-band modeling method. The DNNs consisted of a gated linear unit [34] including the sigmoid activation

layer and tanh activation layer, and batch normalization [35] was carried out before applying each activation function. Adam [36] was used as the optimization method. During training, the cepstrum of the source and target speech was normalized to have zero mean and unit variance. The batch size and number of epochs were set to 1,000 and 100, respectively. The model parameters of the DNNs used with the proposed lifter-training method were initialized with the conventional method. The initial value of the lifter coefficient was set to that of the lifter for minimum phasing. For narrow-band VC and full-band VC with our sub-band modeling method, the learning rates for the conventional method and proposed lifter-training method were set to 0.0005 and 0.00001, respectively. When applying the conventional method to full-band VC without our sub-band modeling method, the learning rate was set to 0.0001.

We used an Intel (R) Core i7-6850K CPU @ 3.60 GHz in the evaluation of processing time to show the effectiveness of our online VC system in a CPU environment. We set the weight of vocoder-guided training $\lambda$ and that of GV compensation $\mu$ to 10 and 100, respectively. In the preliminary experiment, we used three methods for data augmentation: pitch shift, time stretch, and time shift referring to Arakawa's study [8]. As a result, the data augmentation did not improve the converted-speech quality in both intra- and cross-gender cases, so we did not apply it in the following evaluations.

### 6.2 Evaluation of Lifter-Training Method

#### 6.2.1 Objective Evaluation

We compared root mean squared error (RMSE) of the proposed lifter-training method and conventional method when changing $l$. We set the truncated tap length $l$ to 128, 64, 48, and 32. The RMSE was obtained by taking the squared root of Eq. (1). Figure 12 shows a plot of the RMSEs in m2m and f2f cases VC using narrow-band speech (16 kHz). The proposed lifter-training method achieved higher-precision conversion than the conventional method for all $l$. The differences in the RMSEs between the proposed and conventional methods also tended to become more significant when $l$ was smaller. This result indicates that the proposed lifter-training method can reduce the effect of filter truncation.

#### 6.2.2 Subjective Evaluation

To investigate the effectiveness of the proposed lifter-training method, we conducted a series of preference AB tests on speech quality and XAB tests on speaker similarity of converted speech. Thirty listeners participated in each of the evaluations through our crowd-sourced evaluation systems, and each listener evaluated ten speech samples. We used a $t$-test with a significance level $\alpha$ of 0.05. The target speaker's natural speech was used as the reference X in the preference XAB tests. We used the same conditions for all the XAB and AB tests.

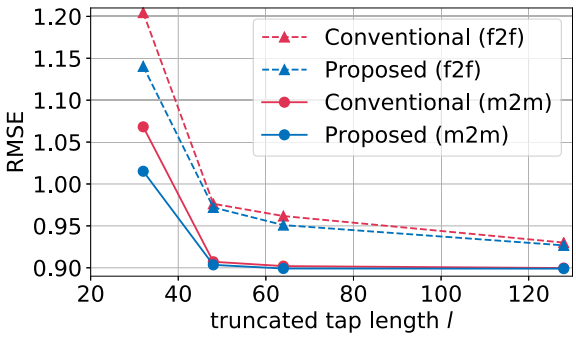


**Fig. 12** RMSEs of our lifter-training ("Proposed") and conventional methods at each $l$ in narrow-band **(16 kHz)** VC.

**Table 1** Preference scores with our lifter-training ("Proposed") and conventional methods in narrow-band case **(16 kHz)**.

(a) **Speaker similarity**

| Spkr | Proposed | Score | p-value | Conventional |
|---|---|---|---|---|
| m2m | $l = 32$ | **0.587** vs. 0.413 | $1.3 \times 10^{-5}$ | $l = 32$ |
| | $l = 32$ | 0.463 vs. 0.537 | $7.3 \times 10^{-2}$ | $l = 512$ |
| | $l = 48$ | 0.533 vs. 0.467 | $1.0 \times 10^{-1}$ | $l = 48$ |
| | $l = 48$ | **0.550** vs. 0.450 | $1.4 \times 10^{-2}$ | $l = 512$ |
| f2f | $l = 32$ | **0.642** vs. 0.358 | $< 10^{-10}$ | $l = 32$ |
| | $l = 32$ | **0.543** vs. 0.457 | $3.4 \times 10^{-2}$ | $l = 512$ |
| | $l = 48$ | **0.613** vs. 0.387 | $1.3 \times 10^{-8}$ | $l = 48$ |
| | $l = 48$ | **0.548** vs. 0.452 | $2.0 \times 10^{-2}$ | $l = 512$ |

(b) **Speech quality**

| Spkr | Proposed | Score | p-value | Conventional |
|---|---|---|---|---|
| m2m | $l = 32$ | **0.687** vs. 0.313 | $< 10^{-10}$ | $l = 32$ |
| | $l = 32$ | 0.529 vs. 0.471 | $2.3 \times 10^{-1}$ | $l = 512$ |
| | $l = 48$ | **0.606** vs. 0.394 | $8.7 \times 10^{-8}$ | $l = 48$ |
| | $l = 48$ | 0.523 vs. 0.477 | $2.6 \times 10^{-1}$ | $l = 512$ |
| f2f | $l = 32$ | **0.807** vs. 0.193 | $< 10^{-10}$ | $l = 32$ |
| | $l = 32$ | **0.742** vs. 0.258 | $< 10^{-10}$ | $l = 512$ |
| | $l = 48$ | **0.581** vs. 0.419 | $5.5 \times 10^{-5}$ | $l = 48$ |
| | $l = 48$ | 0.513 vs. 0.487 | $5.1 \times 10^{-1}$ | $l = 512$ |

In the preliminary experiments, we confirmed that the converted-speech quality with the conventional method significantly deteriorated when we truncate the filter length to 32 and 48. Therefore, we compared several settings of the conventional method and proposed lifter-training method with $l = 32$ and 48. Table 1 lists the results for narrow-band (16 kHz) VC. Compared to the truncated conventional method ("Conventional ($l = 32, 48$)"), we can see that the proposed lifter-training method significantly outperformed the conventional one in terms of speaker similarity and speech quality. Also, compared to the non-truncated conventional method ("Conventional ($l = 512$)"), the proposed lifter-training method ("Proposed ($l = 32, 48$)") had the same or higher quality. These results indicate that the proposed lifter-training method can reduce the tap length to 1/16 without degrading converted-speech quality whereas the truncated conventional method significantly degrades converted-speech quality.

**Table 2** Preference scores with a combination of our methods ("Proposed") and benchmark in full-band **(48 kHz)** VC.

(a) **Speaker similarity**

| Spkr | Proposed | Score | p-value | Benchmark |
|---|---|---|---|---|
| m2m | $l = 32$ | 0.537 vs. 0.463 | $7.3 \times 10^{-2}$ | $l = 2048$ |
| | $l = 48$ | 0.493 vs. 0.507 | $7.4 \times 10^{-1}$ | $l = 2048$ |
| f2f | $l = 32$ | 0.516 vs. 0.484 | $2.5 \times 10^{-1}$ | $l = 2048$ |
| | $l = 48$ | 0.475 vs. 0.525 | $8.3 \times 10^{-2}$ | $l = 2048$ |

(b) **Speech quality**

| Spkr | Proposed | Score | p-value | Benchmark |
|---|---|---|---|---|
| m2m | $l = 32$ | **0.840** vs. 0.160 | $< 10^{-10}$ | $l = 2048$ |
| | $l = 48$ | **0.828** vs. 0.172 | $< 10^{-10}$ | $l = 2048$ |
| f2f | $l = 32$ | **0.810** vs. 0.190 | $< 10^{-10}$ | $l = 2048$ |
| | $l = 48$ | **0.593** vs. 0.407 | $4.2 \times 10^{-6}$ | $l = 2048$ |

### 6.3 Evaluation of Sub-Band Modeling Method

We evaluated a combination of our lifter-training and sub-band modeling methods (hereafter, "sub-band lifter modeling method") in the full-band VC. We defined the conventional method simply extended to full-band VC without our sub-band modeling method (Sect. 2.4) as the benchmark, which was also used in the following sections. The tap length of the differential filter was 2048 in the benchmark. With our method, we truncated the tap length of the filter to 48 and 32. Table 2 shows the results of XAB tests on speaker similarity and AB tests on speech quality. In terms of speaker similarity, there were no significant differences between our method and the benchmark. On the other hand, our method significantly outperformed the benchmark in terms of speech quality. Therefore, we can confirm that our method can improve converted-speech quality while significantly reducing computational cost.

### 6.4 Comparison of Online and Offline VC

To evaluate online conversion, we compared the converted-speech quality of our online VC system described in Sect. 5.1 with that of offline VC described in Sect. 4. As a subjective evaluation, we conducted AB tests on speech quality and XAB tests on speaker similarity. We did not apply pre-emphasis and enhancing techniques described in Sect. 5.2 to the online conversion to compare under fair conditions. Furthermore, we did not truncate the filter in both online and offline conversions because the effect of filter truncation is expected to be the same with both VC methods. Table 3 shows that there is no significant difference between online and offline conversions in terms of both speaker similarity and speech quality. Therefore, we can confirm that online conversion shows the same converted-speech quality as offline conversion.

**Table 3** Preference scores with our online VC system described in Sect. 5.1 and offline VC described in Sect. 4.

(a) **Speaker similarity**

| Spkr | | Score | p-value | |
|---|---|---|---|---|
| m2m | online | 0.493 vs. 0.506 | $7.4 \times 10^{-1}$ | offline |
| f2f | online | 0.486 vs. 0.513 | $5.1 \times 10^{-1}$ | offline |

(b) **Speech quality**

| Spkr | | Score | p-value | |
|---|---|---|---|---|
| m2m | online | 0.517 vs. 0.483 | $4.2 \times 10^{-1}$ | offline |
| f2f | online | 0.490 vs. 0.510 | $6.2 \times 10^{-1}$ | offline |

### 6.5 Computational Complexity and Processing Time of our Online VC System

#### 6.5.1 Computational Complexity

In this section, we estimated the complexity of our online VC systems as an evaluation of computational efficiency. Our online full-band VC system consists of sub-band processing ("Sub-band"), cepstrum analysis ("Cepstrum"), inference with the DNN ("Inference"), the Hilbert transform ("Hilbert trans."), and filtering ("Filtering"). The complexity of each process can be calculated from the parameters in Sect. 6.1. We converted the complexity to floating point operations per second, i.e., FLOPS and considered 0.300 GFLOPS complexity for other neglected calculations ("Other"), e.g., pre-emphasis and F0 transformation. In the same manner, we calculated the complexity of our online narrow-band VC system considering 0.100 GFLOPS for neglected operations.

Table 4 (a) lists the results when the filter was full-tap (512 taps), truncated to 1/4 tap length and truncated to 1/16 tap length in the narrow-band and full-band cases. In the narrow-band case, the total complexity was 0.86 GFLOPS with the 1/4-tap filter and 0.60 GFLOPS with the 1/16-tap filter, whereas the complexity with the full-tap filter was 1.91 GFLOPS. These results indicate that we can significantly reduce complexity by using our lifter-training method with filter truncation and our online narrow-band VC system achieves real-time conversion with a CPU of a single board computer (e.g., Raspberry Pi). In the full-band case, our online VC system attained 2.50 GFLOPS with 1/4-tap filter and can convert full-band speech with lower computational cost than LPCNet [37] for narrow-band (16 kHz) waveform synthesis. Note that the total complexity was around 20 GFLOPS with the benchmark, and the key difference is the filtering operation, which requires around 16.8 GFLOPS with benchmark and can be reduced to around 0.1 GFLOPS with the proposed system. Therefore, we can confirm that filter truncation and sub-band processing can efficiently reduce computational cost. The complexity of sub-band processing is more dominant than complexity reduction with our lifter-training method, but we can further reduce the computational cost of the whole system by incorporating our lifter-training method.

**Table 4** Estimated complexity and measured RTF of our online VC system in narrow-band (16 kHz) and full-band (48 kH) cases.

(a) **Complexity (GFLOPS)**

| Frequency | Tap length | Sub-band | Cepstrum | Inference | Hilbert trans. | Filtering | Other | Total |
|---|---|---|---|---|---|---|---|---|
| Narrow-band | Full-tap | - | 0.043 | 0.330 | 0.041 | 1.399 | 0.100 | 1.91 |
| | 1/4-tap | | | | | 0.350 | | 0.86 |
| | 1/16-tap | | | | | 0.088 | | 0.60 |
| Full-band | Full-tap | 1.430 | 0.043 | 0.330 | 0.041 | 1.399 | 0.300 | 3.54 |
| | 1/4-tap | | | | | 0.350 | | 2.50 |
| | 1/16-tap | | | | | 0.088 | | 2.23 |

(b) **RTF**

| Frequency | Tap length | Sub-band | Cepstrum | Inference | Hilbert trans. | Filtering | Other | Total |
|---|---|---|---|---|---|---|---|---|
| Narrow-band | Full-tap | - | 0.005 | 0.133 | 0.008 | 0.190 | 0.012 | 0.35 |
| | 1/4-tap | | | | | 0.052 | | 0.21 |
| | 1/16-tap | | | | | 0.015 | | 0.17 |
| Full-band | Full-tap | 0.308 | 0.005 | 0.133 | 0.008 | 0.264 | 0.052 | 0.77 |
| | 1/4-tap | | | | | 0.070 | | 0.58 |
| | 1/16-tap | | | | | 0.020 | | 0.53 |

### 6.5.2 Processing Time

To evaluate the computational performance of our online VC systems, we measured the processing time with a single CPU then calculated the real-time factor (RTF) by dividing the average processing time of frames within an utterance by the length of the input waveform (i.e., 5 ms). Table 4 (b) lists the results. In the full-band case, the RTF of our online VC system was 0.77 with the full-tap filter, 0.58 with the 1/4-tap filter, and 0.53 with the 1/16-tap filter, demonstrating that our online full-band VC system can operate in real time. Note that the RTF was around 3.0 with the benchmark method, and we can see that our proposed methods, on which our online full-band VC system is based, can enhance computational efficiency to achieve real-time operation. In this experimental evaluation, our system processed each 25 ms frame within 5 ms. If we need to use a very low-power CPU or change other parameters, it would be necessary to further reduce the RTF by using a larger frame shift (e.g., 10 ms) [38].

## 6.6 Evaluation of Methods for Enhancing our Online VC System

We investigated the effectiveness of the methods presented in Sect. 5.2 through subjective evaluations. Tables 5 and 6 list the evaluation results. In these tables, the columns labeled "EQ", "GV" and "Voc" denote whether we applied F0 equalization (Sect. 5.2.1), GV compensation (Sect. 5.2.3), or vocoder-guided training (Sect. 5.2.2), respectively.

### 6.6.1 F0 Equalization in Pre-Processing

We first evaluated the F0 equalization method described in Sect. 5.2.1. Table 5 shows the results of subjective evaluations. In "EQ" column, "src" indicates F0 equalization that changes the F0 of source speech (Fig. 11 (b)), "tar" denotes F0 equalization that changes the F0 of target speech (Fig. 11 (c)), and blank is correspond to the method without F0 equalization. We compared "src" and "tar" with the method without F0 equalization. In the f2f and m2m cases, i.e., intra-gender conversion, the method without F0 equalization outperformed "tar" in both speaker similarity and speech quality, and F0 equalization reduced the converted-speech quality. On the other hand, in the case of f2m and m2f, i.e., cross-gender conversion, we can see that "tar" outperformed the method without F0 equalization under all conditions. In cross-gender conversion, F0 transformation with PICOLA significantly modifies the spectrum of source speech, and there are larger differences between the source spectrum and target spectrum than in intra-gender cases. Therefore, F0 equalization makes it easier to capture the difference of spectral envelopes for cross-gender VC.

**Table 5** Preference scores when comparing F0 equalization that changed F0 of source speech ("src" in column "EQ") and F0 equalization that changed F0 of target speech ("tar" in column "EQ") with method without F0 equalization (blank in column "EQ").

(a) **Speaker similarity**

| Spkr | EQ | GV | Voc | Score | p-value | EQ | GV | Voc |
|---|---|---|---|---|---|---|---|---|
| m2m | tar | | | 0.381 vs. **0.619** | $1.8 \times 10^{-9}$ | | | |
| | tar | | | 0.410 vs. **0.590** | $1.4 \times 10^{-5}$ | src | | |
| f2f | tar | | | 0.433 vs. **0.567** | $1.1 \times 10^{-3}$ | | | |
| | tar | | | **0.547** vs. 0.453 | $2.2 \times 10^{-2}$ | src | | |
| f2m | tar | | | **0.570** vs. 0.430 | $5.8 \times 10^{-4}$ | | | |
| | tar | | | **0.606** vs. 0.394 | $8.7 \times 10^{-8}$ | src | | |
| m2f | tar | | | **0.577** vs. 0.423 | $1.6 \times 10^{-4}$ | | | |
| | tar | | | **0.616** vs. 0.384 | $3.2 \times 10^{-9}$ | src | | |

(b) **Speech quality**

| Spkr | EQ | GV | Voc | Score | p-value | EQ | GV | Voc |
|---|---|---|---|---|---|---|---|---|
| m2m | tar | | | 0.260 vs. **0.740** | $< 10^{-10}$ | | | |
| | tar | | | 0.273 vs. **0.727** | $< 10^{-10}$ | src | | |
| f2f | tar | | | 0.506 vs. 0.494 | $7.5 \times 10^{-1}$ | | | |
| | tar | | | **0.594** vs. 0.406 | $2.7 \times 10^{-6}$ | src | | |
| f2m | tar | | | **0.603** vs. 0.397 | $3.3 \times 10^{-7}$ | | | |
| | tar | | | **0.679** vs. 0.321 | $< 10^{-10}$ | src | | |
| m2f | tar | | | **0.655** vs. 0.345 | $< 10^{-10}$ | | | |
| | tar | | | **0.670** vs. 0.330 | $< 10^{-10}$ | src | | |

**Table 6** Preference scores with vocoder-guided training and GV compensation.

(a) **Speaker similarity**

| Spkr | EQ | GV | Voc | Score | p-value | EQ | GV | Voc |
|---|---|---|---|---|---|---|---|---|
| m2m | | ✓ | | 0.484 vs. 0.516 | $4.2 \times 10^{-1}$ | | | |
| | | | ✓ | 0.520 vs. 0.480 | $3.3 \times 10^{-1}$ | | | |
| f2f | | ✓ | | 0.457 vs. **0.543** | $3.4 \times 10^{-2}$ | | | |
| | | | ✓ | **0.587** vs. 0.413 | $2.0 \times 10^{-5}$ | | | |
| f2m | tar | ✓ | | **0.577** vs. 0.423 | $1.1 \times 10^{-4}$ | tar | | |
| | tar | | ✓ | **0.547** vs. 0.453 | $2.2 \times 10^{-2}$ | tar | | |
| m2f | tar | ✓ | | **0.590** vs. 0.410 | $9.2 \times 10^{-6}$ | tar | | |
| | tar | | ✓ | **0.617** vs. 0.383 | $7.3 \times 10^{-9}$ | tar | | |

(b) **Speech quality**

| Spkr | EQ | GV | Voc | Score | p-value | EQ | GV | Voc |
|---|---|---|---|---|---|---|---|---|
| m2m | | ✓ | | **0.572** vs. 0.428 | $2.6 \times 10^{-4}$ | | | |
| | | | ✓ | **0.603** vs. 0.397 | $3.3 \times 10^{-7}$ | | | |
| f2f | | ✓ | | **0.565** vs. 0.435 | $1.3 \times 10^{-3}$ | | | |
| | | | ✓ | **0.617** vs. 0.383 | $1.1 \times 10^{-8}$ | | | |
| f2m | tar | ✓ | | 0.513 vs. 0.487 | $5.2 \times 10^{-1}$ | tar | | |
| | tar | | ✓ | **0.593** vs. 0.407 | $4.2 \times 10^{-6}$ | tar | | |
| m2f | tar | ✓ | | **0.652** vs. 0.348 | $1.2 \times 10^{-14}$ | tar | | |
| | tar | | ✓ | **0.752** vs. 0.248 | $< 10^{-10}$ | tar | | |

However, in intra-gender cases, the degradation of training data by DTW and WORLD synthesis is more dominant on converted-speech quality than F0 equalization. Furthermore, the converted-speech quality of "tar" was higher than that of "src" in all the cross-gender cases. This is seemingly because "tar" does not modify source speech in the training data, whereas "src" changes the properties of the source speech used for training and conversion steps. In the following evaluations, we did not apply F0 equalization to the intra-gender conversion and applied "tar" to the cross-gender conversion.

### 6.6.2 Vocoder-Guided Training and GV Compensation

We investigated the effectiveness of vocoder-guided training described in Sect. 5.2.2 and GV compensation described in Sect. 5.2.3. As described at the end of Sect. 6.6.1, we used F0 equalization only in the cross-gender cases. Table 6 lists the results of the subjective evaluations of intra- and cross-gender cases with and without vocoder-guided training and with and without GV compensation. In the intra-gender conversion cases, vocoder-guided training and GV compensation did not improve speaker similarity except for one case. However, in the cross-gender conversion cases, they improved speaker similarity under all conditions. For speech quality, we can see that conversion with vocoder-guided training and GV compensation outperformed that without them. We also conducted an objective evaluation of GV compensation, as shown in Appendix A. The results suggest that GV values tend to move closer to the target GV values by using the compensation method for cross-gender conversion. From the above results, we used only vocoder-guided training in the intra-gender conversion cases and applied both methods to the cross-gender conversion cases in the following evaluations.

**Table 7** Preference scores when comparing speaker similarity of three methods: our online narrow-band VC system incorporating improvements ("Narrow-band+"), benchmark method ("Benchmark"), and our online full-band VC system incorporating improvements ("Full-band+").

| Spkr | | Score | p-value | |
|---|---|---|---|---|
| m2m | Full-band+ | 0.470 vs. 0.530 | $1.4 \times 10^{-1}$ | Benchmark |
| | Full-band+ | 0.513 vs. 0.487 | $5.1 \times 10^{-1}$ | Narrow-band+ |
| f2f | Full-band+ | **0.752** vs. 0.248 | $< 10^{-10}$ | Benchmark |
| | Full-band+ | **0.693** vs. 0.306 | $< 10^{-10}$ | Narrow-band+ |
| f2m | Full-band+ | 0.507 vs. 0.493 | $7.4 \times 10^{-1}$ | Benchmark |
| | Full-band+ | **0.647** vs. 0.353 | $< 10^{-10}$ | Narrow-band+ |
| m2f | Full-band+ | 0.388 vs. **0.612** | $5.7 \times 10^{-9}$ | Benchmark |
| | Full-band+ | 0.450 vs. **0.550** | $1.4 \times 10^{-2}$ | Narrow-band+ |

## 6.7 Comprehensive Evaluation of Our Online VC Systems

In this section, we comprehensively evaluated the converted-speech quality with our online VC systems. We first define each method to be evaluated. "Full-band+" and "Full-band" are versions of our online full-band VC system with and without the improvements mentioned in Sect. 6.6, respectively. "Narrow-band+" is our online narrow-band VC incorporating the methods described in Sect. 5.2 in the same manner as "Full-band+". "Benchmark" is the conventional method implemented in the form of online conversion and simply extended to full-band VC without our sub-band modeling method. We discuss the evaluation of speaker similarity with each method in Sect. 6.7.1 and the MOS evaluation tests for naturalness in Sect. 6.7.2. Audio samples generated with these methods are publicly available for f2f conversion[†].

### 6.7.1 Subjective Evaluation for Speaker Similarity

In Sect. 6.3, we compared our sub-band modeling method with the benchmark, and there were no significant difference between them in terms of speaker similarity in the intra-gender cases. In this section, we first discuss investigating the effectiveness of the methods evaluated in Sect. 6.6 by comparing "Full-band+" with "Benchmark". Furthermore, we explored the effect of the frequency-band extension by comparing "Full-band+" and "Narrow-band+". Table 7 lists the results. In the f2f case, "Full-band+" attained higher speaker similarity than "Benchmark" by introducing the improvements. Furthermore, "Full-band+" showed a higher score than "Narrow-band+", demonstrating the effectiveness of the bandwidth extension. In the m2m and f2m cases, there were no differences between "Full-band+" and "Benchmark", and "Full-band+" significantly outperformed "Narrow-band+". However, in the m2f case, the scores of "Benchmark" and "Narrow-band+" were higher than that with "Full-band+". Future research is needed to investigate the reasons for equal or better performance in the f2f, m2m and f2m cases and lower performance in the m2f case.

[†]https://takaaki-saeki.github.io/rtvc_filter_demo/

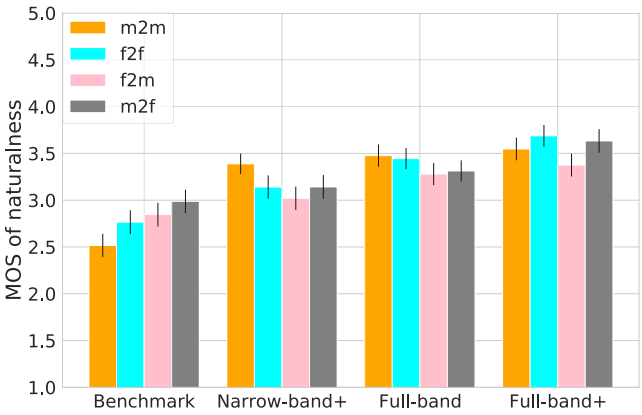


**Fig. 13** MOS scores with our online narrow-band VC system incorporating several methods evaluated in Sect. 6.6 ("Narrow-band+"), the benchmark method defined in Sect. 6.3 ("Benchmark"), our online full-band VC system with basic structures described in Sect. 5.1 ("Full-band") and our online full-band VC system incorporating several improvements ("Full-band+").

#### 6.7.2 MOS Evaluation Test for Naturalness

To evaluate converted-speech quality, we conducted a MOS evaluation test for naturalness of converted-speech. Forty listeners participated in each evaluation through our crowd-sourced evaluation systems, and each listener evaluated 20 speech samples. Figure 13 shows the results, where the error bar means the 95% confidence interval. "Narrow-band+" showed higher naturalness than "Benchmark" despite having a lower sampling frequency than "Benchmark". "Full-band" outperformed "Benchmark" and "Narrow-band+", demonstrating the effectiveness of our sub-band modeling method for the online full-band VC system. Furthermore, the average MOS of "Full-band+" was higher than that of "Full-band" in intra- and cross-gender cases. Our online full-band VC system attained a MOS score of 3.6 of naturalness, whereas it was around 2.8 with the benchmark method and 3.2 with our online narrow-band VC system.

## 7. Conclusion

We proposed two high-fidelity and computationally efficient neural voice conversion (VC) methods based on a direct waveform modification using spectral differentials. First, we proposed a lifter-training method with filter truncation for short-tap filtering. It performed data-driven phase reconstruction by training a lifter for the Hilbert transform considering filter truncation. We then proposed a sub-band modeling for real-time full-band VC. It enhanced computational efficiency by reducing sampling points of signals converted with filtering and improved converted-speech quality by modeling only the low-frequency band that contributes to speaker identity and avoiding high-frequency modeling. Furthermore, we presented the implementation methods of our real-time, online, full-band VC system using only a single CPU for practical applications. Experimental results indicated that our proposed methods significantly improve converted-speech quality and computational efficiency in both narrow-band and full-band cases, and our VC system based on our proposed methods can synthesize full-band converted speech in real time using a low-power CPU and can attain a mean opinion score of 3.6 / 5.0 regarding naturalness.

Even though our current system achieves high speech quality and real-time operation, the speaker similarity is limited due to the simple DFT-based feature analysis. When our current system is applied to a real-world situation, it can perform a rough speaker conversion (e.g., speaker effects) with high speech quality. In future work, we will mainly work on improving the feature analysis part to enhance the speaker similarity. Furthermore, we focused on a real-time VC system that can be applied to relatively limited use cases (e.g., parallel data and one-to-one speaker mapping) in this work. Our additional task would be extending our system to other VC frameworks, including non-parallel training [39], [40] and multi-speaker conversion with speaker adaptation techniques [41].

## Acknowledgements

Part of this work was supported by the MIC/SCOPE #182103104.

### References

[1] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," Proc. ICASSP, New York, U.S.A., pp.655–658, April 1988.
[2] T. Toda, "Augmented speech production based on real-time statistical voice conversion," Proc. GlobalSIP, Atlanta, U.S.A., pp.592–596, Dec. 2014.
[3] Y. Stylianou, O. Cappe, and E. Moulines, "Continuous probabilistic transform for voice conversion," IEEE Transactions on Speech and Audio Processing, vol.6, no.2, pp.131–142, 1998.
[4] T. Toda, A.W. Black, and K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," IEEE Transactions on Audio, Speech, and Language Processing, vol.15, no.8, pp.2222–2235, 2007.
[5] S. Desai, E.V. Raghavendra, B. Yegnanarayana, A.W. Black, and K. Prahallad, "Voice conversion using artificial neural networks," Proc. ICASSP, Taipei, Taiwan, pp.3893–3896, April 2009.
[6] L. Sun, S. Kang, K. Li, and H. Meng, "Voice conversion using deep bidirectional long short-term memory based recurrent neural networks," Proc. ICASSP, Brisbane, Australia, pp.4869–4873, April 2015.
[7] T. Toda, T. Muramatsu, and H. Banno, "Implementation of computationally efficient real-time voice conversion," Proc. INTERSPEECH, Portland, U.S.A., pp.94–97, Sept. 2012.
[8] R. Arakawa, S. Takamichi, and H. Saruwatari, "Implementation of DNN-based real-time voice conversion and its improvements by audio data augmentation and mask-shaped device," Proc. SSW10, Vienna, Austria, pp.93–98, Sept. 2019.
[9] K. Kobayashi, T. Toda, and S. Nakamura, "Intra-gender statistical singing voice conversion with direct waveform modification using log-spectral differential," Speech Communication, vol.99, pp.211–220, 2018.
[10] A. Tamamori, T. Hayashi, K. Kobayashi, K. Takeda, and T. Toda, "Speaker-dependent WaveNet vocoder," Proc. INTERSPEECH,

Stockholm, Sweden, pp.1118–1122, Aug. 2017.
[11] N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimberg, A.v.d. Oord, S. Dieleman, and K. Kavukcuoglu, "Efficient neural audio synthesis," arXiv, vol.abs/1609.03499, 2018.
[12] X. Wang, S. Takaki, and J. Yamagishi, "Neural source-filter-based waveform model for statistical parametric speech synthesis," Proc. ICASSP, Calgary, Canada, pp.5916–5920, April 2018.
[13] S. Imai, K. Sumita, and C. Furuichi, "Mel log spectrum approximation (MLSA) filter for speech synthesis," Electronics and Communications in Japan, vol.66, no.2, pp.10–18, 1983.
[14] H. Suda, G. Kotani, S. Takamichi, and D. Saito, "A revisit to feature handling for high-quality voice conversion," Proc. APSIPA ASC, Hawaii, U.S.A., pp.816–822, Nov. 2018.
[15] S. Takamichi, Y. Saito, N. Takamune, D. Kitamura, and H. Saruwatari, "Phase reconstruction from amplitude spectrograms based on directional-statistics deep neural networks," Signal Processing, vol.169, p.107368, 2020.
[16] R. Crochiere and L. Rabiner, Multirate digital signal processing, Englewood Cliffs, N.J.: Prentice-Hall, 1983.
[17] T. Okamoto, K. Tachibana, T. Toda, Y. Shiga, and H. Kawai, "Subband WaveNet with overlapped single-sideband filterbanks," Proc. ASRU, Okinawa, Japan, pp.698–704, Dec. 2017.
[18] T. Saeki, Y. Saito, S. Takamichi, and H. Saruwatari, "Lifter training and sub-band modeling for computationally efficient and high-quality voice conversion using spectral differentials," Proc. ICASSP, Barcelona, Spain, pp.7784–7788, May 2020.
[19] T. Saeki, Y. Saito, S. Takamichi, and H. Saruwatari, "Real-time, full-band, online DNN-based voice conversion system using a single CPU," Proc. INTERSPEECH, Shanghai, China, pp.1021–1022, Oct. 2020.
[20] T. Fukada, K. Tokuda, T. Kobayashi, and S. Imai, "An adaptive algorithm for mel-cepstral analysis of speech," Proc. ICASSP, San Francisco, U.S.A., pp.137–140, March 1992.
[21] S.-C. Pei and H.-S. Lin, "Minimum-phase FIR filter design using real cepstrum," IEEE Transactions on Circuits and Systems II: Express Briefs, vol.53, no.10, pp.1113–1117, 2006.
[22] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, "Learning representations by back-propagating errors," Nature, vol.323, pp.533–536, 1986.
[23] P.S. Nidadavolu, C. Lai, J. Villalba, and N. Dehak, "Investigation on bandwidth extension for speaker recognition," Proc. INTERSPEECH, Hyderabad, India, pp.1111–1115, Sept. 2018.
[24] H. Miyoshi, Y. Saito, S. Takamichi, and H. Saruwatari, "Voice conversion using sequence-to-sequence learning of context posterior probabilities," Proc. INTERSPEECH, Stockholm, Sweden, pp.1268–1272, Aug. 2017.
[25] T. Kaneko, H. Kameoka, K. Hiramatsu, and K. Kashino, "Sequence-to-sequence voice conversion with similarity metric learned using generative adversarial networks," Proc. INTERSPEECH, Stockholm, Sweden, pp.1283–1287, Aug. 2017.
[26] H. Kameoka, K. Tanaka, D. Kwasny, T. Kaneko, and N. Hojo, "Convs2s-vc: Fully convolutional sequence-to-sequence voice conversion," IEEE Transactions on Audio, Speech, and Language Processing, vol.28, pp.1849–1863, June 2020.
[27] N. Morita and F. Itakura, "Time-scale modification algorithm for speech by use of autocorrelation method and its evaluation," IEICE Technical Report, vol.86, pp.9–16, May 1986.
[28] 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, July 2016.
[29] H. Kawahara, I. Masuda-Katsuse, and A.D. Cheveigné, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," Speech Communication, vol.27, no.3–4, pp.187–207, 1999.
[30] Y. Saito, S. Takamichi, and H. Saruwatari, "Statistical parametric speech synthesis incorporating generative adversarial networks," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol.26, no.1, pp.84–96, Jan. 2018.
[31] S. Takamichi, R. Sonobe, K. Mitsui, Y. Saito, T. Koriyama, N. Tanji, and H. Saruwatari, "Jsut and jvs: free japanese voice corpora for accelerating speech synthesis research," Acoustical Science and Technology, vol.41, pp.761–768, 2020.
[32] y_benjo and MagnesiumRibbon, "Voice-actress corpus." http://voice-statistics.github.io/
[33] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," Proc. KDD, Anchorage, U.S.A., pp.2623–2631, Aug. 2019.
[34] Y.N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," Proc. ICML, Sydney Australia, pp.933–941, Aug. 2017.
[35] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," Proc. ICML, Lille, France, pp.448–456, July 2015.
[36] D. Kingma and B. Jimmy, "Adam: A method for stochastic optimization," arXiv, vol.abs/1412.6980, 2014.
[37] J.-M. Valin and J. Skoglund, "Lpcnet: Improving neural speech synthesis through linear prediction," Proc. ICASSP, Brighton, U.K., pp.5891–5895, May 2019.
[38] Y. He, T.N. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang, Q. Liang, D. Bhatia, Y. Shangguan, B. Li, G. Pundak, K.C. Sim, T. Bagby, S. Chang, K. Rao, and A. Gruenstein, "Streaming end-to-end speech recognition for mobile devices," Proc. ICASSP, Brighton, United Kingdom, pp.6381–6385, May 2019.
[39] T. Kaneko and H. Kameoka, "CycleGAN-VC: Non-parallel voice conversion using cycle-consistent adversarial networks," Proc. EUSIPCO, Rome, Italy, pp.2100–2104, Sept. 2018.
[40] T. Kaneko, H. Kameoka, K. Tanaka, and N. Hojo, "CycleGAN-VC2: Improved CycleGAN-based non-parallel voice conversion," Proc. ICASSP, Brighton, U.K., pp.6820–6824, May 2019.
[41] D.-Y. Wu and H.-Y. Lee, "One-shot voice conversion by vector quantization," Proc. ICASSP, Barcelona, Spain, pp.7734–7738, May 2020.

## Appendix A: Objective Evaluation of Statistical Compensation

In this section, we show results of objective evaluations on statistical compensation described in Sect. 5.2.3. We calculated the average GV values of converted cepstrum features within test utterances for the case with and without the compensation. Figure A· 1 shows the results. As a result, we did not confirm significant improvement in GV values with the statistical compensation method for all the cases.

The subjective evaluations in Sect. 6.6.2 showed that the compensation did not improve the speaker similarity for intra-gender conversion. The results in Fig. A· 1 also shows that some GV values of converted spectra move away from the target GV values by using the compensation method. For cross-gender conversion, low-order (e.g., 0–20 th) GV values of converted cepstrum tend to move closer to that of target cepstrum by using the compensation method, similar to the results of the subjective evaluation in Sect. 6.6.2.

To summarize the results of the objective and subjective evaluations, we can infer the effect of GV compensation in our system is limited. Unlike cepstrum features obtained

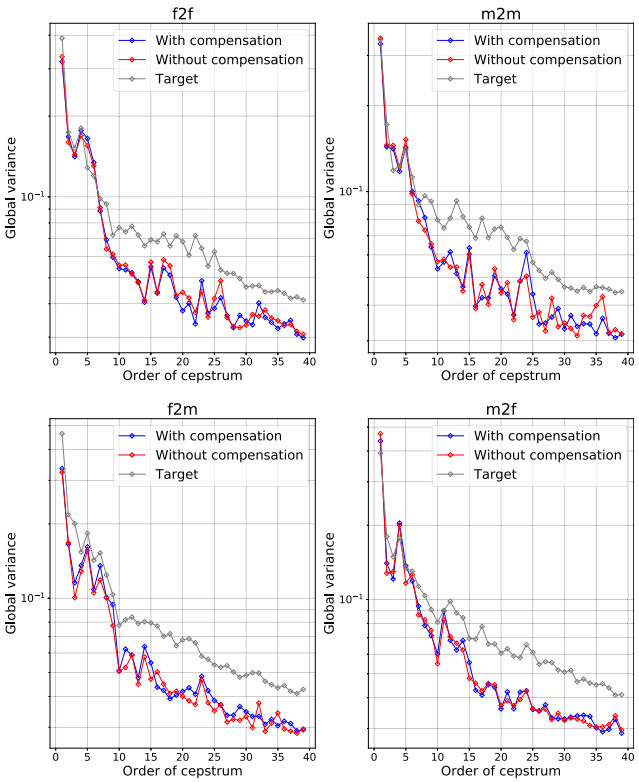


**Fig. A· 1** Average GV values of converted cepstrum within test utterances.

from STRAIGHT/WORLD spectrum, which is used in previous works focusing on GV compensation, DFT-based cepstrum used in this paper more depends on F0. We assume that this caused the limited compensation effect of GV training.

**Takaaki Saeki** received his B.E. degree in engineering from The University of Tokyo, Japan in 2019. He is studying for his M.S. degree in creative informatics at The University of Tokyo. His research interests include voice conversion, text-to-speech synthesis, 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 and a Student Member of IEEE SPS.

**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.

**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.