ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-6327-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICASSP49357.2023.10096901

JSV-VC: JOINTLY TRAINED SPEAKER VERIFICATION AND VOICE CONVERSION MODELS

Shogo Seki, Hirokazu Kameoka, Kou Tanaka, Takuhiro Kaneko

NTT Communication Science Laboratories, NTT Corporation, Japan

ABSTRACT

This paper proposes a variational autoencoder (VAE)-based method for voice conversion (VC) on arbitrary source-target speaker pairs without parallel corpora, i.e., non-parallel any-to-any VC. One typical approach is to use speaker embeddings obtained from a speaker verification (SV) model as the condition for a VC model. However, converted speech is not guaranteed to reflect a target speaker's characteristics in a naive combination of VC and SV models. Moreover, speaker embeddings are not designed for VC problems, leading to suboptimal conversion performance. To address these issues, the proposed method, JSV-VC, trains both VC and SV models jointly. The VC model is trained so that converted speech is verified as the target speaker in the SV model, while the SV model is trained in order to output consistent embeddings before and after the VC model. The experimental evaluation reveals that JSV-VC outperforms conventional any-to-any VC methods quantitatively and qualitatively.

***Index Terms*—** Voice conversion, speaker verification, parallel-data-free, any-to-any mapping, joint learning

1. INTRODUCTION

Voice conversion (VC) is the process of converting the non-linguistic or the para-linguistic information of an input speech while preserving the linguistic information. VCs are helpful in a wide variety of applications, such as speaker identity modification in text-to-speech (TTS) [1], synthetic data augmentation in automatic speech recognition (ASR) [2], and speaking-aid or -assistant systems [3, 4]. Depending on the limitation of the training corpus, VC models fall into either parallel VCs or parallel-data-free (i.e., non-parallel) VCs.

In parallel VCs, the training corpus is restricted to parallel corpora in which both source and target speakers utter each sentence. One widely studied approach to parallel VCs involves training a feature mapping function, represented by a Gaussian mixture model (GMM) [5–7] or a deep neural network (DNN) [8, 9]. In particular, parallel VCs using sequence-to-sequence (S2S) models [10–13] have been shown to provide impressive performance.

On the other hand, non-parallel VCs have also attracted attention due to the ease of data collection, where two main approaches have been proposed; generative adversarial network (GAN) [14]-based approach, and variational autoencoder (VAE) [15]-based approach. A typical GAN-based non-parallel VC employs a CycleGAN [16–20] or a many-to-many extension called StarGAN [21–24]. The conversion network (generator) is adversarially trained with a spoofing detection network (discriminator), where a cycle-consistency loss plays an essential role in preserving the linguistic context. VAE-based non-parallel VCs typically employ a conditional VAE (CVAE) [25–29]. Encoder and decoder networks are trained so that the encoder outputs latent features independent from an input condition while the decoder reconstructs acoustic features conditioned on the given condition. A conversion process can be performed by simply changing the condition. There has been proposed several extensions such as vector quantization (VQ) in a latent space [30], cycle-consistency loss as CycleGAN [31], autoencoder (AE)-based training criterion [32].

Motivated by the great success of non-parallel VCs, recently, various methods [32–35] have attempted to a more general situation, where arbitrary source-target pairs can be applicable, i.e., any-to-any situation. As one of the non-parallel any-to-any VC baselines, AutoVC has been proposed [32]. AutoVC improves VAE-based non-parallel approaches and includes an AE-based VC model with a carefully-designed bottleneck and a pre-trained speaker recognition (SR) model, namely the speaker verification (SV) model. In AutoVC, the SV model is trained using a well-established criterion called generalized end-to-end (GE2E) loss [36] in advance. Then, the VC model is trained using the AE-based training criterion by combining the speaker embeddings obtained from the SV model as the condition. This framework of using both VC and SV models allows us to handle non-parallel any-to-any VCs, demonstrating the conversion performance. However, one drawback in AutoVC would be that since AutoVC uses self-reconstruction loss only in the VC model training, the VC model is trained not considering the conversion process. As a result, the converted speech is not guaranteed to reflect a target speaker's characteristics well. Another limitation is that the speaker embeddings are optimized for the SV model, not the VC model, leading to suboptimal performance.

To address these issues, this paper proposes a non-parallel any-to-any VC method called JSV-VC, jointly-trained speaker verification and voice conversion models. JSV-VC not only uses a self-reconstruction loss but also uses a training objective that considers the conversion process. Furthermore, JSV-VC employs a joint learning approach, where the VC model is trained so that the converted speech is verified as a target speaker in the SV model. Meanwhile, the SV model is trained in order to output consistent speaker embeddings between the utterances before and after the VC model. It is noteworthy that JSV-VC can be viewed as an any-to-any extension of the non-parallel many-to-many VC using an auxiliary classifier variational autoencoder (ACVAE) called ACVAE-VC [29]. In ACVAE-VC, the VC model is trained considering the conversion process through a speaker identification (SI) model represented by the auxiliary classifier. On the other hand, in the proposed JSV-VC, an auxiliary classifier is treated as the SV model and applied for any-to-any settings.

2. CONVENTIONAL METHOD: AUTOVC

Let the acoustic feature and the attribute class label, e.g., speaker identity, be $\mathbf{X}$ and y, respectively. AutoVC is an AE-based VC

This work was partly supported by JST, CREST Grant Number JPMJCR19A3, Japan.

Authorized licensed use limited to: NTT. Downloaded on November 07,2023 at 09:39:01 UTC from IEEE Xplore. Restrictions apply.

method that can handle any-to-any VC problems, and consists of three modules: a content encoder, a decoder, and a pre-trained style encoder. Using network parameters of the content encoder ϕ', the decoder θ', and the style encoder ψ', the VC model is trained by minimizing the following self-reconstructions losses:

$$\mathcal{H}_1 = \mathbb{E}_{y\sim p(y),\mathbf{X}_1,\mathbf{X}_2 \overset{i.i.d.}{\sim} p(\mathbf{X}|y)}[||f_{\theta'}(f_{\phi'}(\mathbf{X}_1), f_{\psi'}(\mathbf{X}_2)) - \mathbf{X}_1||_2^2], \quad (1)$$

$$\mathcal{H}_2 = \mathbb{E}_{(\mathbf{X},y)\sim p(\mathbf{X},y)}[||g_{\theta'}(f_{\phi'}(\mathbf{X}), f_{\psi'}(\mathbf{X})) - \mathbf{X}||_2^2], \quad (2)$$

$$\mathcal{H}_3 = \mathbb{E}_{y\sim p(y),\mathbf{X}_1,\mathbf{X}_2 \overset{i.i.d.}{\sim} p(\mathbf{X}|y)}[||f_{\psi'}(f_{\theta'}(f_{\phi'}(\mathbf{X}_1), f_{\psi'}(\mathbf{X}_2))) - f_{\psi'}(\mathbf{X}_1)||_1], \quad (3)$$

where $f_{\phi'}(\cdot)$, $f_{\theta'}(\cdot)$, and $f_{\psi'}(\cdot)$ are outputs of the content encoder, the decoder, and the style encoder, respectively, and $g_{\theta'}(\cdot)$ is the decoder's intermediate output. Eqs. (1-(3) represent the reconstruction losses in feature and embedding spaces.

3. PROPOSED METHOD: JSV-VC

The proposed JSV-VC employs the same framework as ACVAE-VC [29]. The relationship between the SI model in ACVAE-VC and the SV model in the proposed JSV-VC is described, and then an SV model using GE2E loss is introduced. Furthermore, the joint learning approach for the VC model and the SV model is presented.

3.1. General framework

JSV-VC assumes that the encoder distribution $q_\phi(\mathbf{Z}|\mathbf{X}, y)$ and the decoder distribution $p_\theta(\mathbf{X}|\mathbf{Z}, y)$ follow Gaussian distributions:

$$q_\phi(\mathbf{Z}|\mathbf{X}, y) = \mathcal{N}(\mu_\phi(\mathbf{X}, y), \mathrm{diag}\sigma_\phi^2(\mathbf{X}, y)), \quad (4)$$

$$p_\theta(\mathbf{X}|\mathbf{Z}, y) = \mathcal{N}(\mu_\theta(\mathbf{Z}, y), \mathrm{diag}\sigma_\theta^2(\mathbf{Z}, y)), \quad (5)$$

where $\mu_\phi(\mathbf{X}, y)$ and $\sigma_\phi^2(\mathbf{X}, y)$ are the encoder outputs, and $\mu_\theta(\mathbf{Z}, y)$ and $\sigma_\theta^2(\mathbf{Z}, y)$ are the decoder outputs. Similar to ACVAE-VC [29], using both encoder and decoder network parameters ϕ and θ, the following variational lower bound to be maximized is used for the training criterion:

$$\mathcal{I} = \mathbb{E}_{(\mathbf{X},y)\sim p(\mathbf{X},y)}[\mathbb{E}_{\mathbf{Z}\sim q_\phi(\mathbf{Z}|\mathbf{X},y)}[\log p_\theta(\mathbf{X}|\mathbf{Z}, y)] - \mathcal{D}_{\mathrm{KL}}[q_\phi(\mathbf{Z}|\mathbf{X}, y)||p(\mathbf{Z})]], \quad (6)$$

where $\mathcal{D}_{\mathrm{KL}}[\cdot||\cdot]$ is the Kullback-Leibler (KL) divergence. We assume the prior distribution $p(\mathbf{Z})$ as a standard Gaussian distribution.

Different from AutoVC [32] and conventional VAE-based VCs [16, 30, 31], the proposed framework incorporates the expectation of the mutual information $I(y; \mathbf{X}|\mathbf{Z})$ into the training criterion. This makes the decoder output $\mathbf{X} \sim p_\theta(\mathbf{X}|\mathbf{Z}, y)$ as correlated as possible with the attribute class label y. Since it is difficult to use the mutual information directly, the following variational lower bound instead is used:

$$\mathcal{J} = \mathbb{E}_{(\mathbf{X}_\mathrm{s},y_\mathrm{s})\sim p(\mathbf{X},y),\mathbf{Z}\sim q_\phi(\mathbf{Z}|\mathbf{X}_\mathrm{s},y_\mathrm{s})}[\mathbb{E}_{(\mathbf{X}_\mathrm{t},y_\mathrm{t})\sim p(\mathbf{X},y),\mathbf{X}\sim p_\theta(\mathbf{X}|\mathbf{Z},y_\mathrm{t})}[\log r_\psi(y_\mathrm{t}|\mathbf{X})]], \quad (7)$$

where $r_\psi(y|\mathbf{X})$ is an auxiliary classifier distribution with the network parameter ψ. In eq. (7), the auxiliary classifier only takes converted features obtained through the encoder network and the decoder network, resulting in insufficient performance. Thus, the following cross-entropy is also incorporated in the training:

$$\mathcal{K} = \mathbb{E}_{(\mathbf{X},y)\sim p(\mathbf{X},y)}[\log r_\psi(y|\mathbf{X})]. \quad (8)$$

3.2. Introduction of speaker verification using GE2E loss

While ACVAE-VC deals with many-to-many VC problems, JSV-VC handles any-to-any VC problems, where source speakers and target speakers can be known or unknown. AutoVC and its variants [35] successfully handle this by introducing speaker representation, i.e., speaker embeddings obtained from SV models using GE2E loss. Following the same methodology, JSV-VC extends the ACVAE-VC framework by introducing an SV model. The connection between the SI model in ACVAE-VC and the SV model in JSV-VC can be derived as follows.

Assume that the auxiliary classifier is composed of a feature extraction network and a linear classifier and a dataset contains K-class samples. Given M training samples $\{(\mathbf{X}_m, y_m)\}_m$ at each training step, eq. (8) can be approximated as follows:

$$\mathcal{K} = \frac{1}{M}\sum\nolimits_m \log r_\psi(y_m|\mathbf{X}_m) = \frac{1}{M}\sum\nolimits_m [r_{m,y_m} - \log\sum\nolimits_k \exp r_{m,k}], \quad (9)$$

where $r_{m,k}$ denotes the unnormalized log probability referred to as "logit", and is defined as:

$$r_{m,k} = \mathbf{w}_k^\mathsf{T}\mathbf{e}_m + b_k, \quad (10)$$

where $\mathbf{w}_k \in [\mathbf{w}_1^\mathsf{T}, \ldots, \mathbf{w}_K^\mathsf{T}]^\mathsf{T} (= \mathbf{W})$, $b_k \in [b_1, \ldots, b_K] (= \mathbf{b})$ are learnable weight and bias. $\mathbf{e}_m = f_\psi(\mathbf{X}_m)$ is the intermediate output of the auxiliary classifier, i.e., speaker embedding.

Since eq. (9) takes the same form as [36], GE2E loss can be introduced by modifying the minibatch building way and the logit. First, M training samples can be constructed by collecting I utterances from J different speakers: $M = IJ$. When the m-th training sample corresponds to the sample from the i-th utterance of the j-th speaker, the index m can be represented as (i, j). Next, the logit $r_{m,k} (= r_{i,j,k})$ in eq. (9) is modified as follows:

$$r_{i,j,k} = \begin{cases} w\cos(\mathbf{e}_{i,j}, \mathbf{c}_j^{\setminus i}) + b & (k = j) \\ w\cos(\mathbf{e}_{i,j}, \mathbf{c}_k) + b & (\text{otherwise}) \end{cases}, \quad (11)$$

where w and b are learnable parameters, and $\mathbf{c}_k = \frac{1}{I}\sum_i \mathbf{e}_{i,k}$ and $\mathbf{c}_j^{\setminus i} = \frac{1}{I-1}\sum_{i'\neq i} \mathbf{e}_{i',j}$ are the centroids including and excluding i-th utterance, respectively. Note that K represents the number of speakers in each minibatch. Similarly, eq. (7) can also be approximated in the same fashion. Thus, the auxiliary classifier can be treated as the SV model in JSV-VC, where speaker embedding $f_\psi(\mathbf{X})$ is used instead of y as the condition of the VC model.

3.3. Joint learning scheme

Since JSV-VC uses speaker embeddings as the condition of the VC model, training criteria eqs. (6, 7) are modified as follows:

$$\mathcal{I} = \mathbb{E}_{\mathbf{X}\sim p(\mathbf{X})}[\mathbb{E}_{\mathbf{Z}\sim q_\phi(\mathbf{Z}|\mathbf{X},f_\psi(\mathbf{X}))}[\log p_\theta(\mathbf{X}|\mathbf{Z}, f_\psi(\mathbf{X}))] - \mathcal{D}_{\mathrm{KL}}[q_\phi(\mathbf{Z}|\mathbf{X}, f_\psi(\mathbf{X}))||p(\mathbf{Z})]], \quad (12)$$

$$\mathcal{J} = \mathbb{E}_{(\mathbf{X}_\mathrm{s},y_\mathrm{s})\sim p(\mathbf{X},y),\mathbf{Z}\sim q_\phi(\mathbf{Z}|\mathbf{X}_\mathrm{s},f_\psi(\mathbf{X}_\mathrm{s}))}[\mathbb{E}_{(\mathbf{X}_\mathrm{t},y_\mathrm{t})\sim p(\mathbf{X},y),\mathbf{X}\sim p_\theta(\mathbf{X}|\mathbf{Z},f_\psi(\mathbf{X}_\mathrm{t}))}[\log r_\psi(y_\mathrm{t}|\mathbf{X})]]. \quad (13)$$

Eq. (12) is currently the training criterion for both the VC model and the SV model, and the speaker embeddings are trained by considering a VC problem. Furthermore, in order to make the speaker

Authorized licensed use limited to: NTT. Downloaded on November 07,2023 at 09:39:01 UTC from IEEE Xplore. Restrictions apply.

embeddings of before and after the VC model more consistent, JSV-VC maximizes the following cosine similarity criterion:

$$\mathcal{L} = \mathbb{E}_{\mathbf{X}_\mathrm{b}\sim p(\mathbf{X})}[\mathbb{E}_{\mathbf{Z}\sim q_\phi(\mathbf{Z}|\mathbf{X}_\mathrm{s},f_\psi(\mathbf{X}_\mathrm{b}))}[\mathbb{E}_{\mathbf{X}_\mathrm{a}\sim p_\theta(\mathbf{X}|\mathbf{Z},f_\psi(\mathbf{X}_\mathrm{b}))}[\cos(f_\psi(\mathbf{X}_\mathrm{b}), f_\psi(\mathbf{X}_\mathrm{a}))]]]. \tag{14}$$

3.4. Conversion procedure

At the conversion, a source feature $\mathbf{X}$ can be converted by using the source and target speaker embeddings $\mathbf{e}_\mathrm{s}$ and $\mathbf{e}_\mathrm{t}$:

$$\hat{\mathbf{X}} = \mu_\theta(\mu_\phi(\mathbf{X}, \mathbf{e}_\mathrm{s}), \mathbf{e}_\mathrm{t}). \tag{15}$$

The speaker embeddings of source and target speakers can be obtained in various ways. In particular, in a one-shot situation in which one source utterance $\mathbf{X}_\mathrm{s}$ and one target utterance $\mathbf{X}_\mathrm{t}$ are only available, the source and target speaker embeddings $\mathbf{e}_\mathrm{s}$ and $\mathbf{e}_\mathrm{t}$ can be obtained using the SV model. Converted feature are then reconstructed to time-domain signals with a neural vocoder [37, 38].

4. EXPERIMENTAL EVALUATION

4.1. Experimental configurations

The experimental evaluation was conducted under an any-to-any and one-shot scenario, where source speakers and/or target speakers are known and/or unknown. The experiment used the CMU Arctic database [39], which consists of recordings of 18 speakers reading phonetically balanced English sentences. Since part of the sentences were not read, the same 592 sentences all the speakers read were used in the experiment, which amounts to 10656 utterances. The sentences were divided into 16, 64, and 512 for testing, validation, and training. For testing, male speaker "rms" and female speaker "slt" were selected as the unknown speakers, and different male speaker "bdl" and female speaker "clb" were selected as the known speakers. For training, "bdl", "clb," and the rest of the 14 speakers, including nine male speakers and file female speakers, were used. This resulted in 8192 training utterances from 16 speakers and 64 test utterances from 4 speakers. All the speech signals were sampled at 16 kHz, and 80-dimensional log mel-spectrograms were extracted with a 64 ms frame length and an 8 ms frameshift.

For comparison, we used as the baseline and compared with JSV-VC. We also investigated AutoVC and JSV-VC with and without joint learning of the SV model. Table. 1 shows a summary of the methods in the experimental evaluation. The VC model in JSV-VC was based on the official implementation of ACVAE-VC[1], which consists of a convolutional neural network (CNN)-based encoder-decoder architecture. The encoder and decoder networks consisted of three-layer convolutional and decovolutional architectures with gated linear units (GLUs) [40], where the dimensions of hidden and latent features were set to 64 and 16, respectively. Similarly, the

Table 1: Categorization of methods for comparison.

Method	Training objective
AutoVC [32]	$\min_{\phi',\theta'} \mathcal{H}_1 + \mathcal{H}_2 + \mathcal{H}_3$
AutoVC w/ SV	$\min_{\phi',\theta',\psi'} \mathcal{H}_1 + \mathcal{H}_2 + \mathcal{H}_3 - \mathcal{K}$
JSV-VC w/o SV	$\max_{\phi,\theta} \mathcal{I} + \mathcal{J} + \mathcal{L}$
JSV-VC (proposed)	$\max_{\phi,\theta,\psi} \mathcal{I} + \mathcal{J} + \mathcal{K} + \mathcal{L}$

[1] https://github.com/kamepong/ACVAE-VC

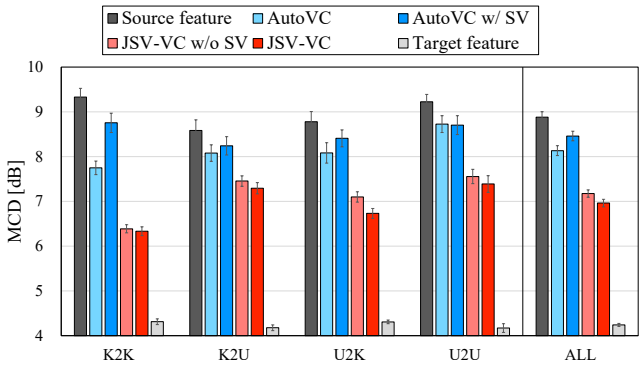

Fig. 1: MCDs with 95 % confidence intervals, where "K2K", "K2U", "U2K", "U2U" represents known-to-known, known-to-unknown, unknown-to-known, and unknown-to-unknown VC settings, respectively.

VC model of AutoVC was prepared using official implementation[2]. Note that the VC model of AutoVC was initialized with a distributed pre-trained model to make the training stable. Following [36], the same SV model was prepared for both AutoVC and JSV-VC ($\psi' = \psi$), which is composed of a three-layer long-short term memory with projection (LSTMP) followed by a fully connected layer and an L2 normalization layer. The dimensions of hidden features, projections, and embeddings were set to 768, 256, and 256, respectively.

For the VC model trainings, the Adam optimizers [41] were used, where the learning rates were set at 1.0×10^{-4} for AutoVC and 1.0×10^{-3} for JSV-VC. Each SV model was trained using a stochastic gradient descent (SGD), where the learning rate was set at 1.0×10^{-2}. The gradient norm of the VC model in JSV-VC and the SV models were clipped, where the clipping values were set to 1.0 for the VC model and 3.0 for the SV models. In the SV models, the gradient scalings for the projection node in LSTMP and GE2E loss were also applied, and the scaling values were set at 0.50 and 0.01, respectively. The weight and bias parameters in GE2E loss were set to 10 and -5, respectively. Each minibatch was built using four utterances from 16 different speakers, where each utterance was trimmed to a clip with a random frame length of 0.8-1.2 seconds. All the models were trained for 500k iterations.

To avoid a combinatorial explosion in converting the test utterances, the sentences of the utterances from source and target speakers were restricted to be identical. This resulted in 192 source-speaker utterance pairs. Speaker embeddings of source and target speakers were extracted with a sliding window approach, where a 1.0 s window length and a 0.2 s frameshift were used. For the generation of time-domain signals, the HiFi-GAN vocoder was used [38]. HiFi-GAN vocoder was prepared from a publicly available implementation[3], where "V2" network architecture was used.

4.2. Objective evaluation

As the evaluation metrics, the average of the Mel-cepstral distortions (MCDs) between the converted and target signals was used in the objective evaluation, where the dynamic time warping (DTW) was applied to align Mel-cepstral sequence pairs in advance. The frame-level MCDs were averaged to obtain the utterance-level MCDs for each converted signal. In addition to AutoVC-based and JSV-VC-based methods, synthesized signals obtained from source features and target features were also evaluated.

[2] https://github.com/auspicious3000/autovc
[3] https://github.com/kan-bayashi/ParallelWaveGAN

Authorized licensed use limited to: NTT. Downloaded on November 07,2023 at 09:39:01 UTC from IEEE Xplore. Restrictions apply.

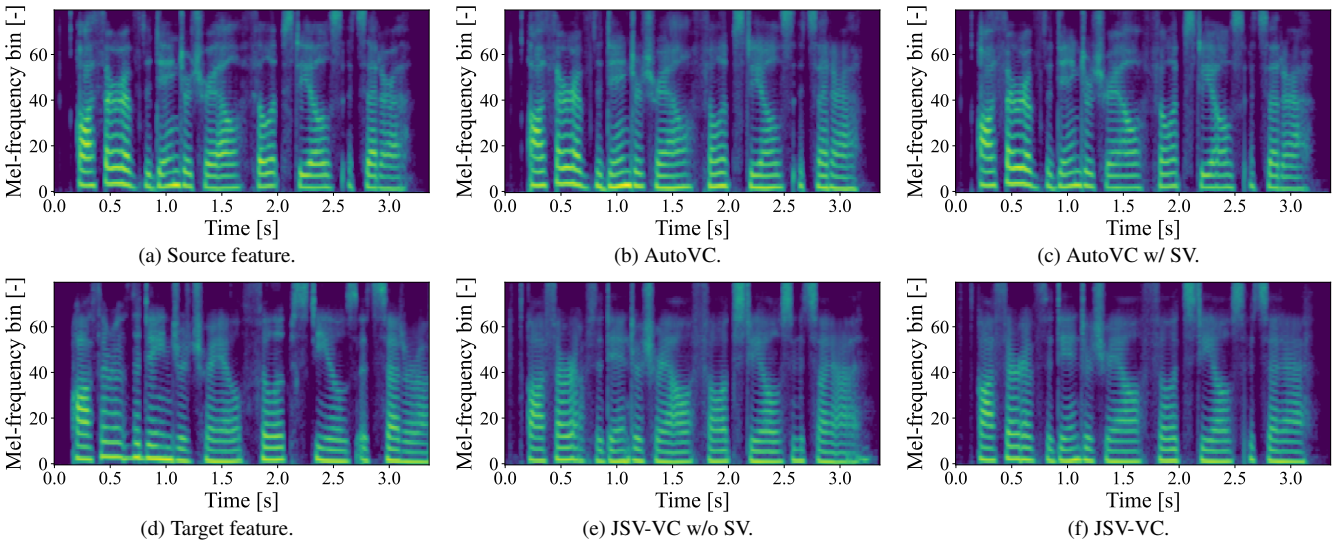

Fig. 2: Mel-spectrograms of (a) source feature, (b) AutoVC, (c) AutoVC w/ SV, (d) target feature, (e) JSV-VC w/o SV, and (f) JSV-VC, where the source speaker is an unknown female speaker and the target speaker is a known female speaker.

Fig. 1 shows a comparison of the conversion performance of each method with different source-target pairs. First, the source feature performs worst, and the target feature performs best, showing the lower and upper bound in the HiFi-GAN vocoder. On the one hand, comparing AutoVC-based methods, the AutoVC with the joint learning causes performance degradation. On the other hand, in a comparison of JSV-VC-based methods, we can see that joint learning provides performance improvements. Although AutoVC and JSV-VC use different network architectures, one clear difference is whether there exist training criteria that consider a conversion process. From these results, we can conclude that it is important to use not only self-reconstruction training objectives but also the training criterion that considers a conversion process, and joint learning contributes to improving conversion performances in the JSV-VC framework.

4.3. Subjective evaluation

A subjective evaluation test on speaker similarity was conducted to investigate the perceptual quality. In the subjective evaluation, five converted samples per source-target pair were used to reduce the evaluation cost, resulting in 60 samples for each method. A preference test was conducted for speaker similarity, where six different conversion methods including synthesized signals from source features and target features were evaluated. Ten subjects were joined; each subject was presented with two utterances and asked to assign a score by selecting "1: Different (sure)", "2: Different (not sure)", "3: Same (not sure)", or "4: Same (sure)". Note that one of two utterances for each sample was the natural speech of a target speaker, and the speaker similarity against the target natural speech was investigated.

Fig. 2 shows a comparison of the converted samples of each method. We can see that the samples from AutoVC-based methods closely resemble the source feature, which implies the conversion failure. On the other hand, JSV-VC-based methods can successfully generates mel-spectrograms more resembling to the target feature. Fig. 3 shows a comparison of the preference scores of each method on speaker similarity. When comparing AutoVC-based methods and JSV-VC methods, the proposed approaches outperform conventional

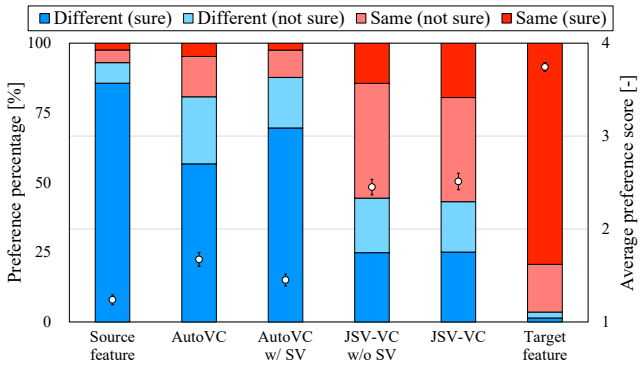

Fig. 3: Preference percentages for speaker similarity, where the average preference scores are denoted as white dots with 95 % confidence intervals.

approaches. It can be seen that AutoVC w/ SV underperforms AutoVC and resembles the result of the source feature, and we can confirm that the conversion performances would be insufficient. Finally, compared to JSV-VC-based methods, JSV-VC has slightly better performance than that without the joint learning approach, demonstrating its effectiveness.

5. CONCLUSION

This paper proposed a non-parallel any-to-any VC method called JSV-VC, jointly-trained speaker verification and voice conversion models. JSV-VC was based on ACVAE-VC, where a training criterion considering a conversion process was included. JSV-VC introduced an SV model, and the VC model was trained so that a converted speech is correctly verified as a target speaker. Meanwhile, the SV model was trained in order to output consistent speaker embeddings before and after the VC model. Furthermore, a joint learning approach of training both the VC model and the SV model was presented. The experimental results revealed that JSV-VC outperforms conventional AutoVC, demonstrating the effectiveness of the joint learning approach.

Authorized licensed use limited to: NTT. Downloaded on November 07,2023 at 09:39:01 UTC from IEEE Xplore. Restrictions apply.

6. REFERENCES

[1] Alexander Kain and Michael W Macon, "Spectral voice conversion for text-to-speech synthesis," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 285–288, 1998.

[2] Bao Thai, Robert Jimerson, Dominic Arcoraci, Emily Prud'hommeaux, and Raymond Ptucha, "Synthetic data augmentation for improving low-resource asr," in *IEEE Western New York Image Processing Workshop*, pp. 1–9, 2019.

[3] Alexander B Kain, John-Paul Hosom, Xiaochuan Niu, Jan PH Van Santen, Melanie Fried-Oken, and Janice Staehely, "Improving the intelligibility of dysarthric speech," *Speech Communication*, vol. 49, no. 9, pp. 743–759, 2007.

[4] Keigo Nakamura, Tomoki Toda, Hiroshi Saruwatari, and Kiyohiro Shikano, "Speaking-aid systems using GMM-based voice conversion for electrolaryngeal speech," *Speech Communication*, vol. 54, no. 1, pp. 134–146, 2012.

[5] 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.

[6] Tomoki Toda, Alan W. Black, and Keiichi 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.

[7] Elina Helander, Tuomas Virtanen, Jani Nurminen, and Moncef Gabbouj, "Voice conversion using partial least squares regression," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 5, pp. 912–921, 2010.

[8] Srinivas Desai, E. Veera Raghavendra, B. Yegnanarayana, Alan W. Black, and Kishore Prahallad, "Voice conversion using Artificial Neural Networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 3893–3896, 2009.

[9] Lifa Sun, Shiyin Kang, Kun Li, and Helen Meng, "Voice conversion using deep Bidirectional Long Short-Term Memory based Recurrent Neural Networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4869–4873, 2015.

[10] Kou Tanaka, Hirokazu Kameoka, Takuhiro Kaneko, and Nobukatsu Hojo, "ATTS2S-VC: Sequence-to-sequence Voice Conversion with Attention and Context Preservation Mechanisms," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 6805–6809, 2019.

[11] Hirokazu Kameoka, Kou Tanaka, Damian Kwaśny, Takuhiro Kaneko, and Nobukatsu Hojo, "ConvS2S-VC: Fully Convolutional Sequence-to-Sequence Voice Conversion," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 1849–1863, 2020.

[12] Wen-Chin Huang, Tomoki Hayashi, Yi-Chiao Wu, Hirokazu Kameoka, and Tomoki Toda, "Pretraining Techniques for Sequence-to-Sequence Voice Conversion," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 745–755, 2021.

[13] Hirokazu Kameoka, Wen-Chin Huang, Kou Tanaka, Takuhiro Kaneko, Nobukatsu Hojo, and Tomoki Toda, "Many-to-Many Voice Transformer Network," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 656–670, 2021.

[14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative Adversarial Nets," *Advances in Neural Information Processing Systems*, vol. 27, pp. 1–9, 2014.

[15] Diederik P Kingma and Max Welling, "Auto-Encoding Variational Bayes," in *International Conference on Learning Representations*, pp. 1–14, 2014.

[16] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks," in *IEEE International Conference on Computer Vision*, pp. 2223–2232, 2017.

[17] Takuhiro Kaneko and Hirokazu Kameoka, "CycleGAN-VC: Parallel-Data-Free Voice Conversion Using Cycle-Consistent Adversarial Networks," in *European Signal Processing Conference*, pp. 2100–2104, 2018.

[18] Takuhiro Kaneko, Hirokazu Kameoka, Kou Tanaka, and Nobukatsu Hojo, "CycleGAN-VC2: Improved CycleGAN-based Non-parallel Voice Conversion," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 6820–6824, 2019.

[19] Takuhiro Kaneko, Hirokazu Kameoka, Kou Tanaka, and Nobukatsu Hojo, "CycleGAN-VC3: Examining and Improving CycleGAN-VCs for Mel-spectrogram Conversion," in *Interspeech*, pp. 2017–2021, 2020.

[20] Takuhiro Kaneko, Hirokazu Kameoka, Kou Tanaka, and Nobukatsu Hojo, "MaskCycleGAN-VC: Learning Non-Parallel Voice Conversion with Filling in Frames," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5919–5923, 2021.

[21] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo, "StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8789–8797, 2018.

[22] Hirokazu Kameoka, Takuhiro Kaneko, Kou Tanaka, and Nobukatsu Hojo, "StarGAN-VC: non-parallel many-to-many Voice Conversion Using Star Generative Adversarial Networks," in *IEEE Spoken Language Technology Workshop*, pp. 266–273, 2018.

[23] Takuhiro Kaneko, Hirokazu Kameoka, Kou Tanaka, and Nobukatsu Hojo, "StarGAN-VC2: Rethinking Conditional Methods for StarGAN-Based Voice Conversion," in *Interspeech*, pp. 679–683, 2019.

[24] Hirokazu Kameoka, Takuhiro Kaneko, Kou Tanaka, and Nobukatsu Hojo, "Non-parallel Voice Conversion With Augmented Classifier Star Generative Adversarial Networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2982–2995, 2020.

[25] Diederik P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling, "Semi-Supervised Learning with Deep Generative Models," in *Advances in Neural Information Processing Systems*, pp. 3581–3589, 2014.

[26] Chin-Cheng Hsu, Hsin-Te Hwang, Yi-Chiao Wu, Yu Tsao, and Hsin-Min Wang, "Voice Conversion from Non-Parallel Corpora Using Variational Auto-encoder," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, pp. 1–6, 2016.

[27] Yuki Saito, Yusuke Ijima, Kyosuke Nishida, and Shinnosuke Takamichi, "Non-Parallel Voice Conversion Using Variational Autoencoders Conditioned by Phonetic Posteriorgrams and D-Vectors," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5274–5278, 2018.

[28] Wen-Chin Huang, Hsin-Te Hwang, Yu-Huai Peng, Yu Tsao, and Hsin-Min Wang, "Voice Conversion Based on Cross-Domain Features Using Variational Auto Encoders," in *International Symposium on Chinese Spoken Language Processing*, pp. 51–55, 2018.

[29] Hirokazu Kameoka, Takuhiro Kaneko, Kou Tanaka, and Nobukatsu Hojo, "ACVAE-VC: Non-Parallel Voice Conversion With Auxiliary Classifier Variational Autoencoder," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 9, pp. 1432–1443, 2019.

[30] Kazuhiro Kobayashi, Wen-Chin Huang, Yi-Chiao Wu, Patrick Lumban Tobing, Tomoki Hayashi, and Tomoki Toda, "crank: An Open-Source Software for Non-parallel Voice Conversion Based on Vector-Quantized Variational Autoencoder," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5934–5938, 2021.

[31] Patrick Lumban Tobing, Yi-Chiao Wu, Tomoki Hayashi, Kazuhiro Kobayashi, and Tomoki Toda, "Non-Parallel Voice Conversion with Cyclic Variational Autoencoder," in *Interspeech*, pp. 674–678, 2019.

[32] Kaizhi Qian, Yang Zhang, Shiyu Chang, Xuesong Yang, and Mark Hasegawa-Johnson, "AutoVC: Zero-Shot Voice Style Transfer with Only Autoencoder Loss," in *International Conference on Machine Learning*, pp. 5210–5219, 2019.

[33] Ju chieh Chou and Hung-Yi Lee, "One-Shot Voice Conversion by Separating Speaker and Content Representations with Instance Normalization," in *Interspeech*, pp. 664–668, 2019.

[34] Yen-Hao Chen, Da-Yi Wu, Tsung-Han Wu, and Hung-yi Lee, "Again-VC: A One-Shot Voice Conversion Using Activation Guidance and Adaptive Instance Normalization," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5954–5958, 2021.

[35] Ruitong Xiao, Haitong Zhang, and Yue Lin, "DGC-Vector: A New Speaker Embedding for Zero-Shot Voice Conversion," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 6547–6551, 2022.

[36] Li Wan, Quan Wang, Alan Papir, and Ignacio Lopez Moreno, "Generalized End-to-End Loss for Speaker Verification," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4879–4883, 2018.

[37] Ryuichi Yamamoto, Eunwoo Song, and Jae-Min Kim, "Parallel WaveGAN: A Fast Waveform Generation Model Based on Generative Adversarial Networks with Multi-Resolution Spectrogram," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 6199–6203, 2020.

[38] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae, "HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis," in *Advances in Neural Information Processing Systems*, pp. 17022–17033, 2020.

[39] John Kominek and Alan W Black, "The CMU Arctic speech databases," in *ISCA Speech Synthesis Workshop*, pp. 223–224, 2004.

[40] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin, "Convolutional Sequence to Sequence Learning," in *International Conference on Machine Learning*, pp. 1243–1252, 2017.

[41] Diederik P Kingma and Jimmy Lei Ba, "Adam: A Method for Stochastic Gradient Optimization," in *International Conference on Leanrning Representations*, pp. 1–15, 2015.

Authorized licensed use limited to: NTT. Downloaded on November 07,2023 at 09:39:01 UTC from IEEE Xplore. Restrictions apply.