2021 IEEE Spoken Language Technology Workshop (SLT) | 978-1-7281-7066-4/20/$31.00 ©2021 IEEE | DOI: 10.1109/SLT48900.2021.9383498

HOW FAR ARE WE FROM ROBUST VOICE CONVERSION: A SURVEY

Tzu-hsien Huang, Jheng-hao Lin*, Hung-yi Lee*

College of Electrical Engineering and Computer Science, National Taiwan University, Taiwan

ABSTRACT

Voice conversion technologies have been greatly improved in recent years with the help of deep learning, but their capabilities of producing natural sounding utterances in different conditions remain unclear. In this paper, we gave a thorough study of the robustness of known VC models. We also modified these models, such as the replacement of speaker embeddings, to further improve their performances. We found that the sampling rate and audio duration greatly influence voice conversion. All the VC models suffer from unseen data, but AdaIN-VC is relatively more robust. Also, the speaker embedding jointly trained is more suitable for voice conversion than those trained on speaker identification.

***Index Terms*—** voice conversion, speaker verification, speaker identification, speaker representation, speaker embedding, network robustness

1. INTRODUCTION

Voice conversion (VC) techniques aim to convert the speaker characteristic of an utterance into that of the target speaker while preserving linguistic content information. In previous work, paired data of speakers were required to achieve VC. Recently, several models were proposed to utilize non-parallel data [1, 2, 3, 4, 5]. DGAN-VC [6] learns disentangled content and speaker information by adversarial training. StarGAN-VC [7] adopts conditional input to achieve many-to-many speaker voice conversion. However, both are restricted to performing VC among seen speakers during training. Zero-shot approaches [8, 9, 10, 11, 12] are then considered, where the models can perform VC among any speakers without fine-tuning. AdaIN-VC [13] applies instance normalization to meet this requirement. AUTOVC [14] employs pretrained d-vector [15] and information bottleneck for this purpose.

Although VC techniques were getting more powerful recently, most of the papers train and evaluate their VC models on the same corpora (usually the VCTK Corpus [16]) with similar recording conditions. However, in real applications, the recording conditions of the utterances can be very different from the training data. Are the VC models nowadays robust enough for real-world applications? The answer is probably no. Huang *et al.* [17] successfully performed adversarial attack on VC models, and this suggests the VC models nowadays may still be not robust enough. However, we do not know how non-robust they are and what kinds of mismatch have impacts on them?

This is probably the first paper studying the robustness of VC models. We survey the robustness of three popular VC models with five frequently-used datasets in the following three aspects.

- How models perform with audios with different sampling rate, duration, or even unseen language.
- To figure out which modules of these models are crucial to voice conversion through ablation study.
- To examine the influence of speaker embeddings so as to figure out which embedding is suitable for VC.

2. VOICE CONVERSION

Here we first introduce VC models and speaker embeddings involved in this paper. We surveyed models achieving disentangled speaker timbre and content information: DGAN-VC [6], AdaIN-VC [13], and AUTOVC [14]. All these models are publicly available and thus can be reproduced easily. For speaker embeddings, we surveyed i-vector [18], d-vector [15], and x-vector [19], all of which perform well in speaker verification, and further introduced a new embedding, v-vector.

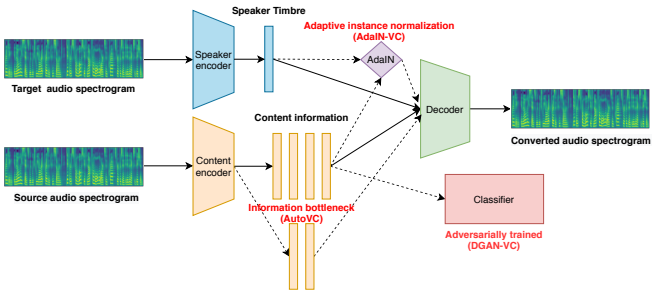


Fig. 1: General framework of voice conversion models.

2.1. Voice conversion models

Recent VC models follow a general encoder-decoder architecture shown in Fig. 1, in which timbre and linguistic content are

* These authors contributed equally.

extracted by speaker encoder (E_s) and content encoder (E_c) respectively. Decoder (D) then construct a converted signals conditioned on representations given by E_s and E_c.

DGAN-VC [6] adopts a two-stage training framework for many-to-many voice conversion. E_s provides one-hot vectors as speaker embeddings to represent different speakers. In the first stage, an additional classifier is applied, taking the content representation as input and trained adversarially so as to obtain speaker-independent linguistic information. In the second stage, a generative adversarial network (GAN) [20] is then adopted to make the output spectra more realistic.

AdaIN-VC [13] uses a variational autoencoder [21] (VAE), where latent representation is restricted by KL divergence loss, for content encoder and employs adaptive instance normalization [22] (AdaIN) to achieve zero-shot VC. Instance normalization used in E_c removes timbre information while preserving the content information, and timbre information is then given by E_s to the decoder with AdaIN layer.

AUTOVC [14] is an autoencoder with a carefully-designed information bottleneck trained with self-reconstruction loss only. The information bottleneck is the dimension of the latent vector between E_c and D. With the bottleneck, linguistic content and timbre information are disentangled without explicit constraints, and the latter is provided by a pretrained d-vector during decoding.

2.2. Speaker embeddings

Speaker embeddings, or speaker representations, are widely used in speaker verification and voice conversion, four types of which are involved in this paper.

In i-vector [18], the speaker and the utterance-dependent GMM supervector $\boldsymbol{M}$ are modeled as

$$\boldsymbol{M} = \boldsymbol{m} + \boldsymbol{T}\boldsymbol{w} \tag{1}$$

where $\boldsymbol{m}$ denotes the speaker-independent GMM supervector and $\boldsymbol{T}$ is a low-rank total variability matrix which captures both speaker and utterance variability. And i-vector is simply the Maximum a Posterior point estimate of $\boldsymbol{w}$.

D-vector [15] applies a DNN as speaker feature extractor and is trained on speaker identification task or with GE2E loss [23]. The last hidden layer's output is taken as a compact representation of a speaker, referred to as d-vector.

X-vector [19] uses a time-delayed neural network (TDNN) to learn temporal context information. It is trained on speaker identification task, and the output of the last hidden layer is taken as the representation of a speaker, called x-vector.

Additionally, v-vector is a new concept introduced in this paper, which is defined as the representation from speaker encoder jointly trained with VC models.

3. EXPERIMENTAL SETTINGS

3.1. Datasets

We used CSTR VCTK Corpus [16], LibriTTS Corpus [24], LibriSpeech ASR Corpus [25] (LibriSpeech), CMU-ARCTIC databases [26] (CMU) and THCHS-30 [27] to evaluate performance of VC models.

For CSTR VCTK Corpus, we split it into two datasets called **VCTK-train** (99 speakers) and **VCTK-test** (10 speakers). To avoid gender bias, we selected 5 male and 5 female speakers randomly as VCTK-test. For LibriTTS Corpus, we used test-clean subset (**LibriTTS**). For LibriSpeech ASR Corpus, we selected test-clean subset (**LibriSpeech**). Table 1 lists detailed information of each dataset. The abbreviations is used for simplicity in figures in later sections.

Table 1: An overview of the datasets.

Dataset	Abbreviation	Speakers	Language	Sample rate
VCTK-train	S	99	En	48k
VCTK-test	U	10	En	48k
LibriTTS	LT	39	En	24k
LibriSpeech	LS	40	En	16k
CMU	C	18	En	16k
THCHS-30	T	60	Zh	16k

3.2. Objective Metrics

We consider two objective metrics: character error rate (CER) and speaker verification accept rate (SVAR). The SVAR is the ratio of the number of utterances accepted by a speaker verification system to the total number of utterances. A lower CER signifies a better content preserved, while a higher SVAR represents a more successful conversion of timbre.

Automatic speech recognition (ASR) measures how well linguistic information from the source utterance is preserved in the converted one. We used Speech-to-Text service in google cloud speech API to compute CER. As for Chinese, we decomposed the Chinese words into pinyin (spelled sounds) to avoid homophone problem, where several characters may pronounce the same but differ in meaning.

Speaker verification (SV) measures whether the converted utterance belongs to the speaker providing timbre information in VC. We used a third-party pretrained speaker encoder[1] to extract speaker embeddings from utterances. A converted utterance is considered successfully converted if the cosine similarity of embeddings between the converted utterance and the target utterance (which provides timbre information in VC) is greater than a given threshold. We obtained the threshold by computing equal error rate (EER) on datasets we used to test. For each speaker, we randomly sampled 128 positive samples and 128 negative samples, and the total number of examples

[1] https://github.com/resemble-ai/Resemblyzer

was more than 100k. The EER is 5.65% and the threshold is 0.6597.

3.3. Implementation details

AUTOVC[2], AdaIN-VC[3] and DGAN-VC[4] used here were obtained from official implementation and were all trained on VCTK-train set. In their original implementations, DGAN-VC and AdaIN-VC utilized Griffin-Lim algorithm [28] to generate waveform from spectrogram while AUTOVC applied WaveNet [29] as vocoder. To eliminate the influence of different vocoders, we modified each model to output 80-dim mel-spectrograms and adopted a pretrained MelGAN [30] as vocoder to convert those mel-spectrograms into waveforms. As DGAN-VC was limited to perform VC only among speakers seen during training in its original implementation, we replaced speaker embeddings of DGAN-VC with the speaker encoder architecture utilized in AdaIN-VC, which can encode utterance from unseen speaker, to meet zero-shot setting.

For speaker embeddings, i-vector and x-vector were obtained from the Kaldi [31] pretrained systems trained on VoxCeleb1 [32] and Voxceleb2 [33]. As for d-vector, we made use of the pretrained d-vector model of AUTOVC trained on VoxCeleb1 and LibriSpeech. For v-vector, we use the pretrained speaker encoder from AdaIN-VC trained on VCTK-train.

4. ROBUSTNESS OF VOICE CONVERSION MODELS

In this section, we discuss the robustness of VC models in different scenarios. In Section 4.1, we investigated the situations that the training and testing utterances are from the same coups (VCTK) but with mismatches from different aspects including speakers, audio sample rates, and utterance durations. In Section 4.2, we surveyed how models perform with noisy utterances. In Section 4.3, we looked into how VC models perform when training and testing data are from different corpora with different characteristics (e.g., different languages).

4.1. Intra-dataset

4.1.1. Experimental setup

We created two new datasets from VCTK-train to model mismatch conditions. **VCTK-16** (S16) is downsampled from VCTK-train (from 48 kHz to 16 kHz); **VCTK-Long** (SL) is generated by concatenating several utterances of the same speaker in VCTK-train so that the duration is longer than 12 seconds. The models in this subsection were trained on VCTK-train. The models were evaluated under 8 scenarios (4 source sets $\times$ 2 target sets) listed in Table 2. In each scenario, we randomly sampled 1000 source-target utterance pairs from source set and target set to perform voice conversion. As shown in Table 2, X-Y denotes that the source utterance providing linguistic content is from corpus X, while the target utterance providing timbre information is from Y, or the VC models convert an utterance from X to make it sound like produced by the speaker in utterance in Y. For S16-S and S16-U, we used the same pairs as which in S-S and S-U respectively.

Table 2: Scenarios of intra-dataset.

Target \ Source	**S**	**U**	**S16**	**SL**
S	S-S	U-S	S16-S	SL-S
U	S-U	U-U	S16-U	SL-U

4.1.2. Results

The results are shown in Fig. 2. Each point in the figure represents the performance of a VC model under a specific scenario in Table 2. Closer to the upper left corner means the better performance, and the closer distribution of a model means it is more robust, that is, less fluctuation under different scenarios. Considering the performance of AdaIN-VC and AUTOVC, we see that AdaIN-VC is better at converting timbre (higher SVAR), whereas AUTOVC has clearer content (lower CER). DGAN-VC is worse than AUTOVC because they have similar CER, but DGAN-VC has lower SVAR.

For the robustness of the models, we see the results of AdaIN-VC cluster together, which suggests it is robust to both mismatched sampling rate and utterance duration. As for DGAN-VC and AUTOVC, the figure shows a mismatched sample rate may be harmful to both ASR and SV results. Longer duration is beneficial for the conversion of speaker timbre with increasing SVAR, but it also increases CER. The influence of sample rate and utterance duration is especially remarkable for AUTOVC. The observations suggest that both the sample rate and audio duration affect voice conversion, which should be noted.

4.2. Noise

4.2.1. Experimental setup

We used the same pairs as which in S-S in Table 2 and added the noises both on the source and target audios. We applied three noises: pink noise [34], brownian noise, and indoor noise. The pink and brownian noises are both generated at the level of roughly -30 dB, while we recorded indoor noise in our laboratory. Ground-truth audios added with noises were also evaluated on both SV and ASR to discover whether objective metrics are robust to these noises.

4.2.2. Results

The results are shown in Fig. 3. The Ground truth in the figure performed well on both CER and SVAR, whose results are

[2]https://github.com/auspicious3000/autovc
[3]https://github.com/jjery2243542/adaptive_voice_conversion
[4]https://github.com/jjery2243542/voice_conversion

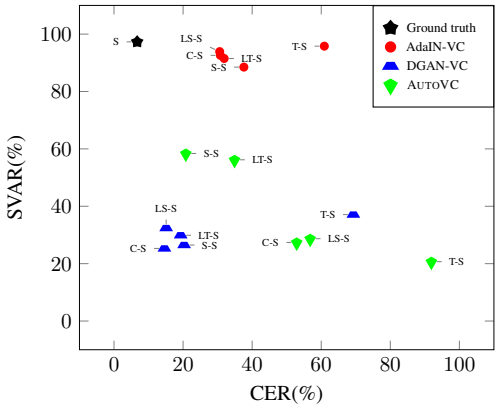


Fig. 2: Conversion results of intra-dataset. $S16$ and SL represent VCTK-16 and VCTK-Long respectively.

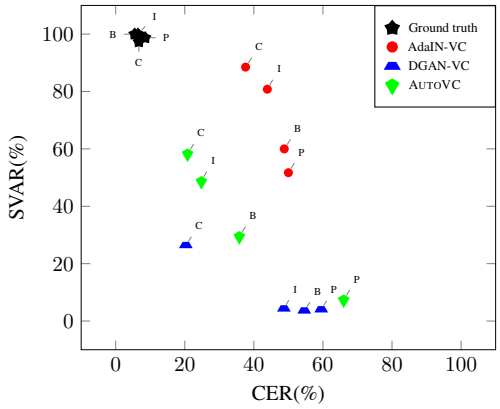


Fig. 3: Conversion results under different noise scenarios. C, B, P and I represent clean audio, brownian noise, pink nosie and indoor noise respectively.

at the upper-left corner of Fig. 3, suggesting that our objective metrics are resistant to these noises. The performance of AdaIN-VC and AUTOVC became worse with brownian noise and pink noise, and the degradation of AUTOVC was more serious than that of AdaIN-VC. In addition, pink noise is especially harmful to AUTOVC. The conversion of DGAN-VC failed under all the noisy inputs. The results suggest AdaIN-VC is more robust to the noises than the others.

4.3. Inter-dataset

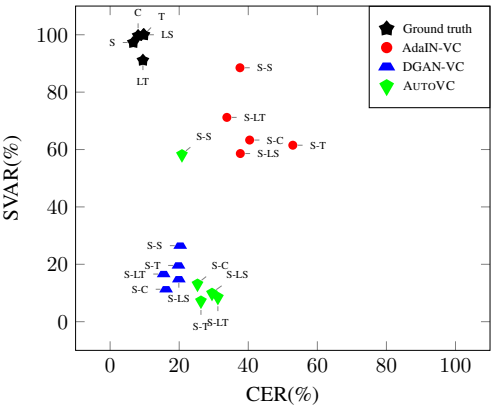
4.3.1. Experimental setup

Here we utilized five datasets, including VCTK-train, LibriTTS, LibriSpeech, CMU, and THCHS-30. The experiment was conducted in two cases: conversion from VCTK-train to the others and conversion from the others to VCTK-train.

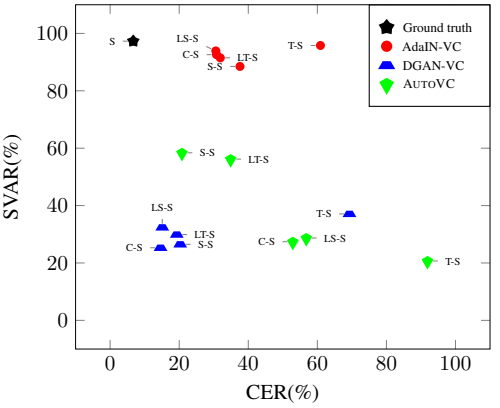
4.3.2. Results

The results are shown in Fig. 4. There is little difference between the performance of DGAN-VC and AUTOVC when conversion is performed between different datasets. However, the performance of AdaIN-VC degrades slightly. Fig. 4b suggests that the performance of AdaIN-VC is stable with unseen speakers as source speaker, while the performance of AUTOVC becomes worse drastically.

The results suggest that the carefully-designed information bottleneck in AUTOVC disentangles content and speaker information successfully on the training set; however, it may not be well-generalized to unseen data. The policy of tuning the size of the information bottleneck to make AUTOVC robust is a critical question to be explored in the future. The results of both from THCHS-30 to VCTK-train and from VCTK-train to THCHS-30 are quite different from others with higher CER and SVAR for all models. This is due to the fact that THCHS-30 (Chinese) and VCTK-train (English) are in different languages, and the capability of cross-lingual VC is still limited for current VC models.

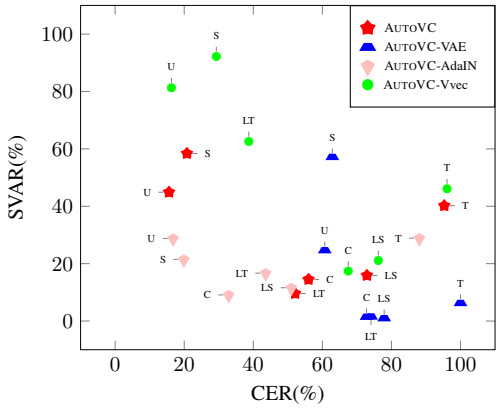


(a) VCTK-train to all datasets

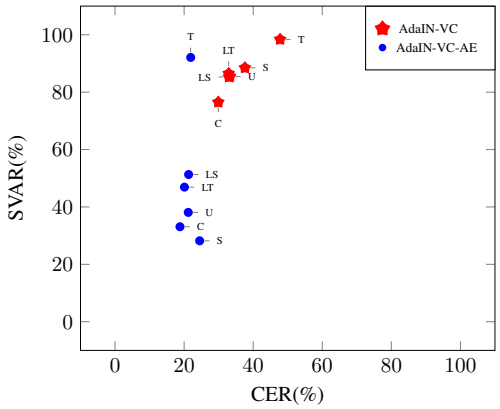


(b) All datasets to VCTK

Fig. 4: Conversion results of inter-dataset.

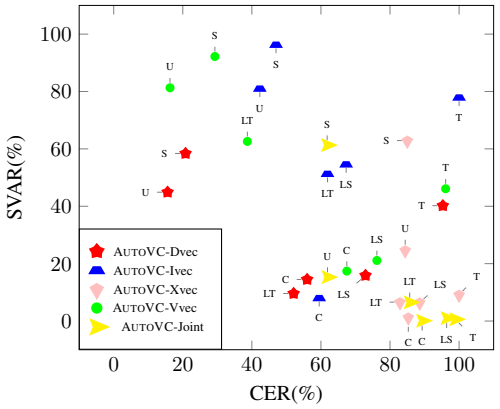


(a) AUTOVC modifications

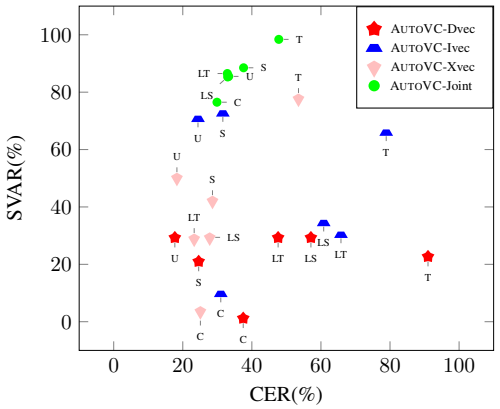


(b) AdaIN-VC modifications

Fig. 5: Conversion results of various combinations of components.



(a) AUTOVC modifications



(b) AdaIN-VC modifications

Fig. 6: Conversion results of different speaker embeddings.

5. INFLUENCE OF MODEL COMPONENTS

In general, AdaIN-VC and AUTOVC performed relatively better than DGAN-VC, and both have their own strengths. We thus further study the two models in this section.

5.1. Experimental setup

First, we trained AUTOVC whose speaker encoder is replaced with v-vector from trained AdaIN-VC (**AUTOVC-Vvec**). Second, we added VAE architecture into AUTOVC (**-VAE**). Third, We integrated AdaIN layer into the decoder of AUTOVC (**-AdaIN**). Finally, we replaced the VAE architecture in AdaIN-VC with an autoencoder one (**AdaIN-VC-AE**).

5.2. Results

The results are shown in Fig. 5. AUTOVC-Vvec performed better than AUTOVC in terms of SVAR on all datasets. However, AUTOVC-Vvec did not show significant improvement in terms of CERs compared to AUTOVC, so it is still not robust to unseen corpora. The influence of speaker embeddings is discussed in Section 6.

The performances of AdaIN-VC-AE and AUTOVC-AdaIN were not ideal on both CER and SVAR, suggesting that using AdaIn layer alone is not powerful enough to convert speaker timbre. AUTOVC-VAE performed the worst in the aspect of CER among AUTOVC modifications. Both small size of latent vector and VAE architecture limit the information that the content encoder provides to the decoder. The undesirable result of AUTOVC-VAE might be attributed to insufficient content information induced by an improper combination of latent vector and VAE architecture.

6. INFLUENCE OF SPEAKER EMBEDDINGS

In this section, we discuss how different speaker embeddings influence the performance of VC.

6.1. Experimental setup

We trained models with three different pretrained speaker embeddings, including i-vector, d-vector and x-vector denoted as **-I**, **-D** and **-X** respectively. We denote VC models whose

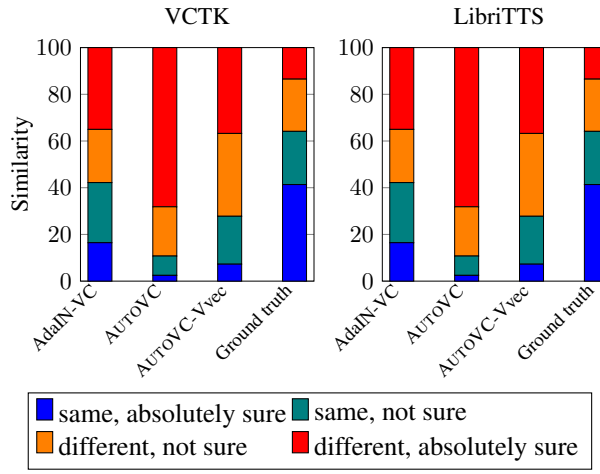


Fig. 7: Subjective evaluation results on speaker similarity of AdaIN-VC, AUTOVC and AUTOVC-Vvec.

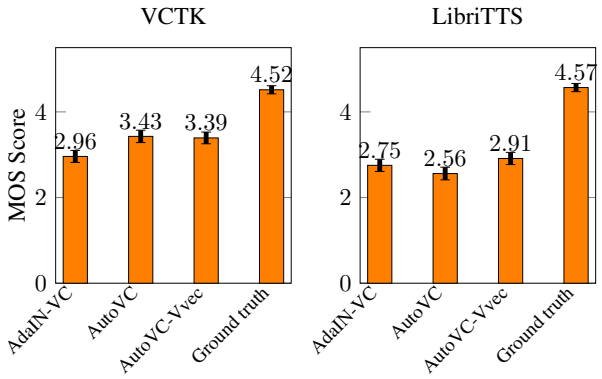


Fig. 8: Subjective evaluation results on naturalness of AdaIN-VC, AUTOVC and AUTOVC-Vvec.

speaker encoder is jointly trained as **-Joint**. In addition, we trained AUTOVC whose speaker encoder is replaced with v-vector from trained AdaIN-VC (**AUTOVC-Vvec**).

6.2. Objective results

The results are shown in Fig. 6. We can see AdaIN-VC-Joint performed the best among all AdaIN-VC modifications, so did AUTOVC-Vvec among all AUTOVC modifications. The models trained with d-vector and x-vector were not capable of converting speaker timbre well. The models trained with i-vector were able to convert timbre well but with unsatisfactory CER. V-vector was trained on far less data than other pretrained speaker embeddings did (VCTK versus VoxCeleb), but still outperformed pretrained ones. Finally, AUTOVC-Joint failed, with extremely high CER and low SVAR in each case.

The results suggest that pretrained speaker embeddings are not as effective for VC as they are in speaker verification although they are pretrained with a large number of speakers. Speaker embedding jointly trained with VC model is more suitable for voice conversion than those trained on speaker verification. Moreover, the results also indicate that the speaker information required for VC and speaker verification are different. Finally, the result also show that AUTOVC needs a pretrained speaker embedding so as to make content encoder able to extract content information only.

6.3. Subjective results

We conducted Mean Opinion Score (MOS) test to rate the quality and similarity of the converted utterances. For speech quality, subjects were asked to score the utterances based on its naturalness from 1 (very bad) to 5 (very good). As for speaker similarity, subjects were given two utterances each time and needed to score based on their similarity from 1 (different, absolutely sure) to 4 (same, absolutely sure). All of the MOS results are reported with 95% confidence intervals.

We compare top three models in the previous experiments, including AdaIN-VC, AUTOVC and AUTOVC with pretrained speaker encoder trained by AdaIN-VC (AUTOVC-Vvec). Models were evaluated on two corpora, VCTK-train and LibriTTS. Both source and target utterances were from the same corpus. For each model, we selected 50 examples from VCTK-train and LibriTTS respectively, and each example was scored by at least 5 subjects.

The results of subjective evaluation are similar to those of objective evaluations, reinforcing our results in the last few sections. Both from the MOS and similarity tests, we can see that AUTOVC-Vvec mitigates the problem that AUTOVC performs undesirably when faced with mismatched data distribution. The results also suggest that utilizing v-vector improves on both objective and subjective metrics.

7. CONCLUSION

We performed a survey on the robustness of current voice conversion models in several perspectives. The sampling rate and audio duration have great impact on these models. In addition, carefully-designed information bottleneck in AUTOVC does not generalize well to unseen data, whereas AdaIN-VC is relatively more robust. Also, speaker embeddings jointly trained with other VC modules are more suitable for the conversion task than those pre-trained for speaker identification. Last but not least, objective and subjective results are highly correlated, indicating that the objective metrics adopted here are appropriate measures for voice conversion.

8. REFERENCES

[1] Li-Wei Chen, Hung-Yi Lee, and Yu Tsao, "Generative adversarial networks for unpaired voice transformation on impaired speech," 2018.

[2] T. Kaneko and H. Kameoka, "Cyclegan-vc: Non-parallel voice conversion using cycle-consistent adversarial networks," in *2018 26th European Signal Processing Conference (EUSIPCO)*, 2018, pp. 2100–2104.

[3] Joan Serrà, Santiago Pascual, and Carlos Segura Perales, "Blow: a single-scale hyperconditioned flow for non-parallel raw-audio voice conversion," in *Advances in Neural Information Processing Systems*, 2019, pp. 6793–6803.

[4] Da-Yi Wu, Yen-Hao Chen, and Hung-Yi Lee, "Vqvc+: One-shot voice conversion by vector quantization and u-net architecture," 2020.

[5] C. Hsu, H. Hwang, Y. Wu, Y. Tsao, and H. Wang, "Voice conversion from non-parallel corpora using variational auto-encoder," in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2016, pp. 1–6.

[6] Ju-chieh Chou, Cheng-chieh Yeh, Hung-yi Lee, and Lin-shan Lee, "Multi-target voice conversion without parallel data by adversarially learning disentangled audio representations," *Proc. Interspeech 2018*, pp. 501–505, 2018.

[7] Hirokazu Kameoka, Takuhiro Kaneko, Kou Tanaka, and Nobukatsu Hojo, "Stargan-vc: Non-parallel many-to-many voice conversion using star generative adversarial networks," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 266–273.

[8] Songxiang Liu, Jinghua Zhong, Lifa Sun, Xixin Wu, Xunying Liu, and Helen Meng, "Voice conversion across arbitrary speakers based on a single target-speaker utterance," in *Proc. Interspeech 2018*, 2018, pp. 496–500.

[9] Andy Liu, Po-chun Hsu, and Hung-yi Lee, "Unsupervised end-to-end learning of discrete linguistic units for voice conversion," 09 2019, pp. 1108–1112.

[10] Kaizhi Qian, Zeyu Jin, Mark Hasegawa-Johnson, and Gautham Mysore, "F0-consistent many-to-many non-parallel voice conversion via conditional autoencoder," 04 2020.

[11] Hui Lu, Zhiyong Wu, Dongyang Dai, Runnan Li, Shiyin Kang, Jia Jia, and Helen Meng, "One-shot voice conversion with global speaker embeddings," 09 2019, pp. 669–673.

[12] Kaizhi Qian, Yang Zhang, Shiyu Chang, David Cox, and Mark Hasegawa-Johnson, "Unsupervised speech decomposition via triple information bottleneck," 2020.

[13] Ju-chieh Chou and Hung-Yi Lee, "One-shot voice conversion by separating speaker and content representations with instance normalization," *Proc. Interspeech 2019*, pp. 664–668, 2019.

[14] 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*, 2019, pp. 5210–5219.

[15] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 4052–4056.

[16] Christophe Veaux, Junichi Yamagishi, Kirsten MacDonald, et al., "Superseded-cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit," 2016.

[17] Chien yu Huang, Yist Y. Lin, Hung yi Lee, and Lin shan Lee, "Defending your voice: Adversarial attack on voice conversion," 2020.

[18] P. Kenny, G. Boulianne, and P. Dumouchel, "Eigenvoice modeling with sparse training data," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 3, pp. 345–354, 2005.

[19] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5329–5333.

[20] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[21] Diederik P Kingma and Max Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[22] Xun Huang and Serge Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1501–1510.

[23] Li Wan, Quan Wang, Alan Papir, and Ignacio Lopez Moreno, "Generalized end-to-end loss for speaker verification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4879–4883.

[24] Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu, "Libritts: A corpus derived from librispeech for text-to-speech," *Proc. Interspeech 2019*, pp. 1526–1530, 2019.

[25] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[26] John Kominek and Alan W Black, "The cmu arctic speech databases," in *Fifth ISCA workshop on speech synthesis*, 2004.

[27] Dong Wang and Xuewei Zhang, "Thchs-30: A free chinese speech corpus," *arXiv preprint arXiv:1512.01882*, 2015.

[28] Daniel Griffin and Jae Lim, "Signal estimation from modified short-time fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, 1984.

[29] Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu, "Wavenet: A generative model for raw audio," in *9th ISCA Speech Synthesis Workshop*, 2016, pp. 125–125.

[30] Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brébisson, Yoshua Bengio, and Aaron C Courville, "Melgan: Generative adversarial networks for conditional waveform synthesis," in *Advances in Neural Information Processing Systems*, 2019, pp. 14910–14921.

[31] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, Jan Silovsky, Georg Stemmer, and Karel Vesely, "The kaldi speech recognition toolkit," 2011, IEEE Catalog No.: CFP11SRW-USB.

[32] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," in *INTERSPEECH*, 2017.

[33] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," in *INTERSPEECH*, 2018.

[34] MICHAEL F. SHLESINGER, "Fractal time and 1/f noise in complex systems," *Annals of the New York Academy of Sciences*, vol. 504, no. 1, pp. 214–228, 1987.