Zero-shot Singing Technique Conversion

Brendan O'Connor[1], Simon Dixon[1], and George Fazekas[1] *

Centre for Digital Music, Queen Mary University of London, UK
b.d.oconnor@qmul.ac.uk

Abstract. In this paper we propose modifications to the neural network framework, AutoVC [17] for the task of singing technique conversion. This includes utilising a pretrained singing technique encoder which extracts technique information, upon which a decoder is conditioned during training. By swapping out a source singer's technique information for that of the target's during conversion, the input spectrogram is reconstructed with the target's technique. We document the beneficial effects of omitting the latent loss, the importance of sequential training, and our process for fine-tuning the bottleneck. We also conducted a listening study where participants rate the specificity of technique-converted voices as well as their naturalness. From this we are able to conclude how effective the technique conversions are and how different conditions affect them, while assessing the model's ability to reconstruct its input data.

Keywords: Voice synthesis, singing synthesis, style transfer, neural network, singing technique, timbre conversion, conditional autoencoder, sequential training, latent loss

1 Introduction

Voice conversion (VC) is the task of converting the timbre of the voice so that the linguistic content is perceived to be spoken by a different person. It has been explored in relation to both singing and speech, which both possess different attributes consideration. Singing voice analysis is considerably more focused on sustained notes, harmonic/rhythmic structure, and relative pitch. In speech, these musical values are non-existent. Instead there is greater emphasis on aperiodic aspects, such as consonant utterances and rapidly shifting spectral envelopes. Tasks like VC and text-to-speech are in far more demand in the industry than singing-related tasks, and have therefore monopolised the spotlight in voice analysis and synthesis research. The latest approaches towards VC achieving state-of-the-art conversions utilise probabilistic machine learning techniques. Public domain speech datasets also vastly overshadow singing datasets in size and availability [11], and so there is still much to be explored in relation to singing analysis and synthesis.

In this paper we tackle the task of singing technique conversion (STC) - the task of converting a singing technique without affecting the perceived identity of the singer, musical structure or linguistic content. We define singing technique as the method of

* This research is funded by the EPSRC and AHRC Centre for Doctoral Training in Media and Arts Technology (EP/L01632X/1).

voice production to achieve different timbres by adjusting the airflow, vocal folds, vocal tract shape, and sympathetic vibrations in the body [5]. We regard STC as a variation of voice conversion (VC), where the possibilities of voice transformation are restricted to be within a realistic variance of timbre for any given singer. We chose the term singing *techniques* as opposed to singing style, due to the latter term's inconsistent use in literature, often referring to a range of very different audio and musical attributes due to its lack of reference to a concrete audio or singing concept.

To achieve STC, we apply a neural network model in the form of the conditioned autoencoder, AutoVC [17]. We discuss certain adaptations made to the architecture and investigate the effects of training it on different permutations of several datasets. To evaluate the model's ability to perform STC, we had participants rate the naturalness of the voice and guess what the target singing technique was supposed to be. Examples of audio used in this listening test can be found online.[1]

Real-time pitch correction algorithms have become commonplace in the music industry and influence the characteristics of modern pop singers today. We believe that the refined task of STC could have a similar influence on music production as it opens up the possibility of artistically manipulating a singer's *performance*, rather than just quantising their pitch. Over the last 5 years, many machine-learning approaches have been proposed to tackle voice transformation for speech (as discussed in the next section), but much less attention has been given to transforming the expression of the singing voice.

2 Related Work

Recent research in VC has been based on neural networks, which have influenced the frameworks proposed in this paper. [15] conditioned an autoencoder (trained on linguistic data) on speaker embeddings generated from a separately trained classifier network. During inference, these embeddings could be replaced to achieve VC. AutoVC [17] adapted this method to work with spectrograms, which will be described in detail in Section 4. This was improved upon by conditioning the network on pitch contours to enforce prosody during conversion [18], and further disentanglement was achieved for timbre, pitch contours, rhythm and utterances simultaneously by utilising 3 separate bottlenecks with different restrictions [19]. [26] achieve VC by using vector quantisation to separate speaker and content information, and later utilised U-nets [21] to compensate for information lost during vector quantisation.

The application of the variational autoencoder (VAE) is well suited for 'many-to-many' conversions (where all examples used for inference are seen during training). [16] use fully convolutional VAEs, conditioned on acoustic features, to perform VC. They combine spectral features of both converted and unconverted reconstructed audio in order to avoid over-smoothing - a known issue with VAEs. While VAEs present an elegant framework, they produce 'blurry' results. Generative Adversarial Networks (GANs) have been known to reproduce better quality reconstructions of images than VAEs. However as they come without an autoencoder they are harder to train and suffer from 'mode collapse', and there has not yet been an elegant proposal for combining

[1] https://github.com/Trebolium/singing_technique_conversion

VAEs with GANs [22]. The use of VAEs has the added benefit of utilising unsupervised learning, which bypasses the issue of low resources regarding labelled singing datasets. [6] used Gaussian-Mixture VAEs (GMVAEs) for controllable speech synthesis, modelling the different attributes of speech as separate prior distributions before combining them in a VAE. For singing voice conversion, [13] adapted AutoVC by conditioning the network on pitch contours transposed to a suitable register for the converted singing, achievable through the implementation of a vocoder. [8] utilised a Wasserstein-GAN framework, using a decoder for pitch contours and another for generating 'formant masks'. The product of these two decoders is the estimated mel-spectrogram for singing. They later explored the capabilities of this framework to achieve timbre and singing style disentanglement [9], where a *singing query* is converted into a singer identity embedding and used to condition both the pitch skeleton and formant-mask encoders on pitch modulation style and singer timbre, respectively. [10] present the only other research we know of that addresses STC. They use GMVAEs to model singer and technique information to perform many-to-many conversions using a VAE architecture that utilises a convolutional recurrent neural network (CRNN) architecture.

The issue remains however, of what can be done with singing datasets which are small and few. [13] notes that the generalisation of the AutoVC framework allows it to be utilised as a Universal Background Model. [1] synthesise monophonic singing datasets by superimposing pitch contours on existing speech datasets. [3] use several autoencoder instances, trained separately on vocoder spectral data and music mixtures, while being conditioned on shared content embeddings and 1-hot speaker embeddings to produce a final network that is singer-independent and generates monophonic singing from musical mixtures. [12] generate novel speaker embeddings by combining embeddings from existing singers as a method of data augmentation.

3 Architecture

We use the AutoVC framework [17] for singing technique transformation, due to its elegant method of applying disentanglement. It is also capable of converting between source and target examples that have not been seen in the training datasets (zero-shot conversion). In AutoVC, a standard autoencoder architecture is conditioned on speaker embeddings that uniquely describe the timbre of a speaker to perform VC on spectrograms. These embeddings are generated by a pretrained speaker verification network [24]. The spectrograms are concatenated with these speaker embeddings, and fed through an encoder E_c, after which the encoded information is again concatenated with speaker embeddings before being fed to the decoder D_c. This conditioning, combined with careful calibration of an appropriate bottleneck size, allows the autoencoder to disentangle speaker timbre from utterance information. AutoVC also contains a 'postnet' convolutional layer which is appended to the decoder to further develop a refined spectrogram from the decoder's output. After training, the speaker embeddings concatenated at the bottleneck can be swapped out to achieve VC. The loss function for AutoVC is a weighted combination of the self-reconstruction loss for both the decoder ($L_{decoder}$) and the postnet ($L_{postnet}$) output spectrograms, and the latent loss (L_{latent}). The latent loss represents the difference between the bottleneck's embedding $E_c(x)$ for the input x

and its reconstructed form $E_c(\hat{x})$. This is summarised in Equation 1, where μ and λ are empirically determined weights. Further details of AutoVC's architecture are given by [17], which we follow in our implementation except for several adjustments discussed in this section.

$$L_{total} = L_{decoder} + \mu L_{postnet} + \lambda L_{latent}. \quad (1)$$

We will herein refer to our implementation of AutoVC as AutoSTC to reflect its purpose of STC. To facilitate this, we developed our own singing technique encoder (STE) to replace the external speaker encoder that was used in the original implementation. The STE is initially trained as a classifier. It takes a mel spectrogram as input, which is split into chunks of 0.5 seconds. These are fed in parallel through a neural network consisting of four 2D-convolutional layers (each of which is followed by batch normalisation, ReLU activation and max-pooling), two dense layers, two BLSTMs, a simplified attention mechanism [20], two more dense layers and finally a classification layer. This architecture was adapted from the VAE used by [10] and influenced by [4]. This network is able to achieve 86% accuracy when classifying singing techniques within a test set of VocalSet (detailed in Section 4, while our implementation of a 1D convolutional network on the waveform data as described by [25] only scored 57%. During conversion, the STE's embedding preceding the classification layer are used for concatenation and conditioning with AutoVC as described above in place of the external speaker encoder embeddings.

4 Training and Inference

The Vocalset dataset [25] used to train the STE consists of recordings of 20 singers performing several musical exercises with different singing techniques. We chose a subset containing the techniques *belt, straight, vibrato, lip trill, vocal fry* and *breathy*, trimming off excess files that appear in one class but not the other, to yield a balanced class subset of 1182 examples (roughly 8K seconds). As the dataset is so small we only partition it into training and test sets by 8:2.

As [13] showed that the sequence of training of different datasets is important, AutoSTC was trained using subsets taken from VocalSet, VCTK [23] and the raw singer recordings from MedleyDB [2] in various permutations. All data was sampled at 16kHz and transformed into 80-bin mel spectrograms. While being trained on one dataset, AutoSTC was simultaneously tested on test sets from all three datasets in between training iterations (the VocalSet test set was the same set omitted when training the STE). We recorded the number iterations and loss values for each dataset where the loss showed no further improvement into Table 1, and transferred the saved neural network parameters of a nearby checkpoint to the proceeding dataset training session in the sequence. We trained AutoSTC once for every permutation of the datasets. Table 1 shows that the order in which datasets are fed to the network does have a considerable impact on its loss. The paths `Vc->Vs->Md` (spanning 750k training steps) and `Vc->Md->Vs` (500k steps) led to the lowest loss values for MedleyDb and VocalSet reconstruction respectively, and were used to train models that generated the examples used in our listening test (see Section 5).

Table 1. Shows losses and training iterations (in parentheses) for VocalSet (left) and MedleyDB (right) along alternative paths. The optimum training path is highlighted in bold from left to right. Training that leads to an increase in loss is indicated with a circumflex, at which point that path is abandoned. For space, the dataset names are shortened as follows: VCTK:Vc, VocalSet:Vs, MedleyDB:Md.

Loss-Iteration for Vs					
Vc	**0.0653(300k)**	Vs	0.0274(100k)	Md	^
		Md	**0.0386(150k)**	**Vs**	**0.0268(50k)**
Vs	0.0347(150k)	Vc	^	Md	-
		Md	^	Vs	-
Md	0.0500(200k)	Vs	0.0290(50k)	Vc	^
		Vc	^	Vc	-

Loss-Iteration for Md					
Vc	**0.0479(500k)**	**Vs**	**0.0474(150k)**	**Md**	**0.0265(100k)**
		Md	0.0295(150k)	Vs	^
Vs	0.0562(150k)	Vc	0.0474(100k)	Md	0.0301(50k)
		Md	0.0370(100k)	Vs	^
Md	0.0367(150k)	Vs	^	Vc	-
		Vc	^	Vc	-

We found the L1 loss between decoder/postnet self-reconstructions and the input encouraged better convergence over L2 loss. We also tested the impact of excluding the latent loss for 100K steps for both VC and STC tasks. Results showed that training without latent loss performs significantly better for both tasks. The loss for STC with latent loss was 0.0237 (and 0.0185 without loss), while spectrograms were blurry and the audio lacked microtonal variation or vibrato, leaving a 'bubbliness' artefact in its absence. Vowels were also not consistently reproduced. These shortcomings however are less noticeable for speech than singing. This result is worth highlighting, as latent loss has been used consistently in frameworks of a similar nature [13, 15, 17].

Further preliminary trials allowed us to fine-tune AutoSTC's bottleneck. We analysed the resynthesised audio and noted that the net size of the feature space was more indicative of audio quality than focusing on time/frequency axis fine-tuning separately. We estimated the threshold to be a downsampling factor of 16 for the time-axis, with each timestep containing 32 features. Lower dimensionality representations resulted in deterioration in the reconstructed audio in a very similar manner to that of the network with latent loss included.

5 Experiment Design

To evaluate our proposed network's ability to perform STC, we conducted a listening study, where 19 participants evaluated the converted audio for specificity and naturalness under the different conditions of models, gender, and source/target techniques used. The first of the models (Vs1) was trained on VocalSet alone and converted Vocalset data. The second (Vs2) was trained using the optimum path for Vocalset presented in Table 1 to also convert VocalSet data, while the third (M1) used the optimum path for MedleyDB to convert MedleyDB data. Converted spectrograms were resynthesised using a pretrained wavenet model provided by the author of the AutoVC paper[2]. Each model produced 8 random examples per participant, while adhering to a balanced representation of both gender and subset (train/test) conditions. To evaluate naturalness, we asked participants to consider how synthetic/realistic the voice itself sounded, and rate them on a scale of 1 (very unnatural) to 5 (very natural). In a separate task to evaluate

[2] https://github.com/auspicious3000/autovc

specificity, participants were given a reference recording featuring a converted singing technique along with 6 unlabelled candidate recordings from the same singer to choose from. These candidate recordings were randomly selected from the relevant dataset partition, so that they each featured 1 of the 6 potential target techniques assigned to the reference recording. Participants were asked to select one recording they thought featured a singing technique closest to that of the reference recording, or more if the answer seemed ambiguous. In the case where reference recordings were converted MedleyDB examples, no ground truth labels existed, and so a singer of the same gender was randomly chosen from Vocalset to represent the 6 candidate singing techniques instead. Each of these tasks was presented 24 times. 6 resynthesised recordings of unconverted audio were also evaluated for naturalness. The interface was built using the Web Audio Evaluation Tool [7].

6 Results

The Mean Opinion Score (MOS) for unconverted data was 3.75 ± 0.34, and is important to consider when analysing the results of the study. This highlights the fact that a considerable amount of perceived naturalness has already been lost during the wavenet resynthesis process, and that the MOS values for technique conversion should be considered with this in mind. It is the comparison between conditions that we are interested in.

To calculate the similarity score S for each condition, we used the formula in Equation 2, where P_n is a binary vector reflecting a participant's true/false predictions (identifying whether each candidate technique was the same as what was presented in the reference audio) for the nth task, C_n is a 1-hot vector reflecting the correct technique for the task, and N is the total number of tasks in the given condition. The similarity score is an average count of correct predictions weighted by the reciprocal of the number of predictions made for the corresponding task.

$$S = \frac{1}{N} \sum_{n=1}^{N} \frac{P_n . C_n}{||P_n||_1}. \tag{2}$$

Figure 1 displays the results obtained from the listening study. The top graph displays MOS values for naturalness, with whiskers indicating the confidence intervals. The lower graph displays similarity scores. The combination of these two graphs give us insight into how each of our models perform, and what conditions influence the naturalness and specificity of the converted singing.

We detected from a Spearman's rank analysis that MOS and similarity scores were not significantly correlated. Similarity scores across all conditions measure higher than the chance level (0.16), which suggests that our models have some success in converting to the target techniques. The condition of source-technique groups does not significantly influence recognisability of the converted singing technique. However, the data would suggest that the features of the target techniques *trill* and *breathy* are significantly more distinguishable than the rest. *Vibrato* scored the lowest for similarity, suggesting that this was a particularly difficult technique to synthesise convincingly. The reason for

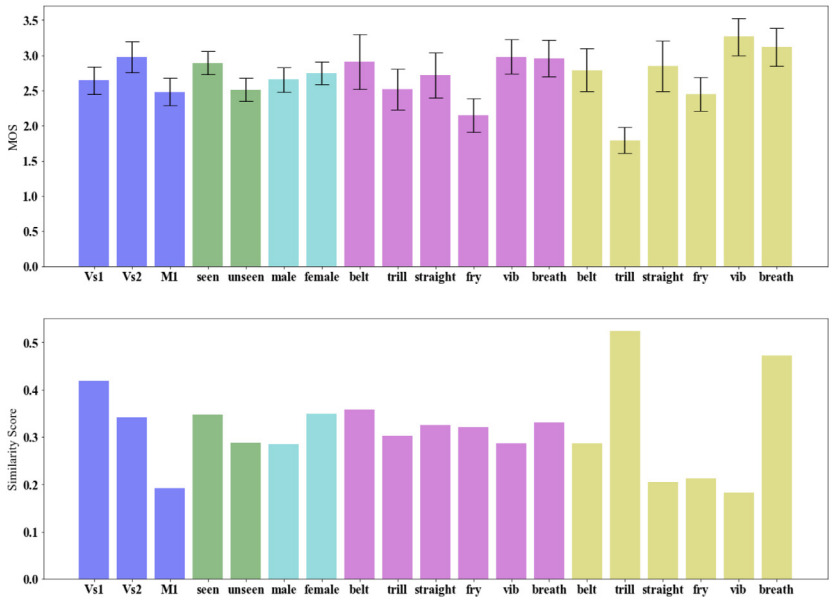

Fig. 1. Top: Bar graph showing naturalness (MOS values and confidence intervals) for all conditions. The colours group together the conditions for (left to right): models, subsets, genders, source technique and target technique. **Bottom:** Bar graph showing similarity scores determined by Equation 2, relative to correct answers.

this is most likely due to the fact that VocalSet, upon which the STE network was trained, contains numerous examples labelled as *belt* and *straight*, while still featuring a considerable amount of frequency modulation (a unique feature of vibrato), making it difficult for AutoSTC to disentangle vibrato from other techniques effectively. It may also be the case that AutoSTC has difficulty disentangling vibrato from pitch contours. Alternatively it is possible that our models instead focused on altering the phonation modes associated with vibrato, which would be considerably less obvious to listeners than identifying whether frequency modulation is occurring.

The inclusion of all datasets in training Vs2 seemingly diminished its ability to accurately convert techniques (although the difference was not statistically significant). The M1 model scored significantly worse than the other models, which tells us that the features learned to generate technique embeddings from the STE network were not generalisable to data outside the dataset the VTE was trained on. There was also no statistically significant difference between gender and subset similarity scores.

In regards to MOS results, the target technique *trill* scored lowest, suggesting that conversions to a trill technique may sound unnatural. Vs2 samples were significantly higher than Vs1 and M1, which suggests that providing the network with multiple datasets does improve its ability to synthesise natural sounding data. The target technique condition *vibrato* scored the highest, but as mentioned above, this may be because the network is making changes more subtle than the frequency modulation which

lessens the amount of transformation required, causing less synthetic artefacts. It is also perfectly possible that participants simply perceive the singing voice to be more natural when vibrato is present.

7 Conclusion

In this paper we have presented a network for vocal technique classification, and the first network to perform zero-shot conversion on singing techniques, achieving above chance level for all tested conditions. We have demonstrated that omitting latent loss and choosing the order in which AutoSTC was fed different datasets significantly diminished its reconstruction loss, improving its ability to reconstruct mel spectrograms. However we can conclude from the results of the listening study that this does not have any significant effect on AutoSTC's ability to perform technique conversion and may even diminish it. We therefore conclude that the features generated by supervised learning on the labelled VocalSet dataset are not sufficient to generalise to recordings of other singers. We also consider that the appearance of frequency modulation in other techniques in VocalSet may have forced the network to give less importance to this vibrato feature (we have however witnessed conversions where frequency modulation was synthesised, but in very limited cases, so we can not rule out the possibility that the AutoSTC framework is incapable of converting singing technique features beyond their spectral filter properties). The findings of our listening study are in agreement the vocal timbre maps generated in our previous research [14].

Augmentation techniques such as those discussed in Section 2 may improve the generalisation of the VTE to unseen data. We would also like to apply the Generalised End-to-End Loss techniques from [24] to the VTE and fine-tune its output embedding size. Due to shortcomings in labelled datasets, we will explore unsupervised/semi-supervised networks such as VAEs. It may also be worth investigating how AutoSTC performs when we condition it on further attributes such as speaker identity, pitch contours and vowel sounds. As we consider STC to be a restricted variation of VC and the fact that there are considerably larger datasets for speech, it may also be worth exploring the effects of pre-training an AutoVC framework for VC before switching its speaker encoder for the singing technique encoder and training it for STC. In future work we will also consider alternative options to the speech-trained wavenet vocoder as this has introduced artefacts to the audio that likely lowered MOS ratings for all audio. We have also observed that AutoSTC was unintentionally able to remove vibrato from singing when underfitting, which may be a capability worth fine-tuning in future work.

References

1. Basak, S., Agarwal, S., Ganapathy, S., Takahashi, N.: End-to-end Lyrics Recognition with Voice to Singing Style Transfer. In: 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 266–270 (2021) https://doi.org/10.1109/ICASSP39728.2021.9415096
2. Bittner, R., Salamon, J., Tierney, M., Mauch, M., Cannam, C., Bello, J.: MedleyDB: A Multitrack Dataset for Annotation-intensive MIR Research. In: 15th International Society for Music Information Retrieval Conference, pp. 155–160. (2014)

3. Chandna, P., Blaauw, M., Bonada, J., Gomez, E: Content Based Singing Voice Extraction from a Musical Mixture. In: 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 781-785 (2020)
4. Choi, K., Fazekas, G., Sandler, M., Cho, K.: Convolutional Recurrentneural Networks for Music Classification. In: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2392–2396. (2017) https://doi.org/10.1109/ICASSP.2017.7952585
5. Heidemann, K.: A System for Describing Vocal Timbre in Popular Pong. In: Music Theory Online, vol. 22, (2016) https://doi.org/10.30535/mto.22.1.2
6. Hsu, W.-N., Zhang, Y., Weiss, R. J., Zen, H., Wu, Y., Wang, Y., Pang, R.: Hierarchical Generative Modeling for Controllable Speech Synthesis. In: The International Conference on Learning Representations, p. 27. New Orleans, LA, USA (2019)
7. Jillings, N., Moffat, D., De Man, B., Reiss, J. D.: Web Audio Evaluation Tool: A browse-based listening environment. In: 12th Sound and Music Computing Conference. Maynooth, Ireland. (2015)
8. Lee, J., Choi, H.-S., Jeon, C.-B., Koo, J., Lee, K.: Adversarially Trained End-to-end Korean Singing Voice Synthesis System. arXic preprint arXiv:1908.01919 (2019)
9. Lee, J., Choi, H.-S., Koo, J., Lee, K.: Disentangling Timbre and Singing Style with Multi-Singer Singing Synthesis System. In: 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 7224–7228 (2020) https://doi.org/10.1109/ICASSP40776.2020.9054636
10. Luo, Y.-J., Hsu, C.-C., Agres, K., Herremans, D.: (2019). Singing Voice Conversion with Disentangled Representations of Singer and Vocal Technique Using Variational Autoencoders. arXiv preprint arXiv:1912.02613 (2019)
11. Meseguer-Brocal, G., Cohen-Hadria, A., Peeters, G.: Creating DALI,a Large Dataset of Synchronized Audio, Lyrics, and Notes. In: Transactions of the International Society for Music Information Retrieval, vol. 3 pp. 55-67 (2020) https://doi.org/10.5334/tismir.30
12. Nachmani, E., Wolf, L.: Unsupervised Singing Voice Conversion. arXiv preprint arXiv:1904.06590 (2019)
13. Nercessian, S.: Zero-shot Singing Voice Conversion. In: Proceedings of the International Society for Music Information Retrieval Conference (2020)
14. O'Connor, B., Dixon, S., Fazekas, G.: An Exploratory Study on Perceptual Spaces of the Singing Voice. In: The 2020 Joint AI Conference on Music Creativity, vol. 1, Stockholm, Sweden (2020)
15. Jia, Y., Zhang, Y., Weiss, R. J., Wang, Q., Shen, J., Ren, F., Wu, Y.: Transfer Learning from Speaker Verification to Multispeaker Text-To-Speech Synthesis. arXiv preprint arXiv:1806.04558 (2019)
16. Kameoka, H., Kaneko, T., Tanaka, K., Hojo, N.: ACVAE-VC: Non-parallel many-to-many voice conversion with auxiliary classifier variational autoencoder. arXiv preprint arXiv:1808.05092 (2020)
17. Qian, K., Zhang, Y., Chang, S., Yang, X., Hasegawa-Johnson, M.: AutoVC: Zero-Shot Voice Style Transfer with Only Autoencoder Loss. In: 36th International Conference of Machine Learning, vol. 47, pp. 5210-5219 (2019)
18. Qian, K., Jin, Z., Hasegawa-Johnson, M., Mysore, G. J.: F0-Consistent Many-To-Many Non-Parallel Voice Conversion Via Conditional Autoencoder. In: 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6284–6288 (2020) https://doi.org/10.1109/ICASSP40776.2020.9054734
19. Qian, K., Zhang, Y., Chang, S., Cox, D., Hasegawa-Johnson, M.: Unsupervised speech decomposition via triple information bottleneck. In: 37th International Conference on Machine Learning, vol. 119, pp. 7792–7802 (2020)

20. Raffel, C., Ellis, D. P. W.: Feed-Forward Networks with Attention Can Solve Some Long-Term Memory Problems. arXiv preprint arXiv:1512.08756 (2016)
21. Ronneberger, O., Fischer, P., Brox, T: U-Net: Convolutional Networks for Biomedical Image Segmentation. In: N. Navab, J. Hornegger, W. M. Wells, A. F. Frangi (eds.) Medical Image Computing and Computer-Assisted Intervention, vol. 9351, pp. 234–241. (2015) https://doi.org/10.1007/978-3-319-24574-428
22. Tolstikhin, I., Bousquet, O., Gelly, S., Schoelkopf, B.: Wasserstein Auto-Encoders. arXiv preprint arXiv:1711.01558 (2019)
23. Veaux, C., Yamagishi, J., MacDonald, K.: CSTR VCTK Corpus: English multi-speaker Corpus for CSTR Voice Cloning Toolkit. (2017) https://doi.org/10.7488/ds/2645
24. Wan, L., Wang, Q., Papir, A., Moreno, I. L.: Generalized End-to-End Loss for Speaker Verification. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4879–4883 (2018) https://doi.org/10.1109/ICASSP.2018.8462665
25. Wilkins, J., Seetharaman, P., Wahl, A., Pardo, B. (2018).: VocalSet: A Singing Voice Dataset. 19th International Society for Music Information Retrieval Conference, Paris ,France pp. 468–474 (2018)
26. Wu, D.Y., Chen, Y.H., Lee, H.Y.: One-Shot Voice Conversion by Vector Quantization. In: 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2020)