Joint Workshop for the Blizzard Challenge and Voice Conversion Challenge 2020
30 October 2020, Shanghai, China

FastVC: Fast Voice Conversion with non-parallel data

Oriol Barbany[1,2], *Milos Cernak*[1]

[1]Logitech Europe S.A., 1015, Lausanne, Switzerland
[2]École Polytechnique Fédérale de Lausanne (EPFL), 1015, Lausanne, Switzerland

milos.cernak@ieee.org

Abstract

This paper introduces FastVC, an end-to-end model for fast Voice Conversion (VC). The proposed model can convert speech of arbitrary length from multiple source speakers to multiple target speakers. FastVC is based on a conditional AutoEncoder (AE) trained on non-parallel data and requires no annotations at all. This model's latent representation is shown to be speaker-independent and similar to phonemes, which is a desirable feature for VC systems. While the current VC systems primarily focus on achieving the highest overall speech quality, this paper tries to balance the development concerning resources needed to run the systems. Despite the simple structure of the proposed model, it outperforms the VC Challenge 2020 baselines on the cross-lingual task in terms of naturalness.

Index Terms: Style Transfer, Voice Conversion, Representation Learning, Speech Processing

1. Introduction

The VC task consists of modifying a speech signal uttered by some source speaker as another target speaker uttered it. Cross-lingual VC allows using different source and target languages. In such a task, the linguistic information is preserved, but speaker-dependent features are changed, which requires semantic reasoning about the input signal.

VC is an inherently ill-posed problem; there are multiple correct outputs. Even if it is easy for humans to identify the concepts of content and style (assessed as naturalness and speaker similarity), it is difficult to quantify the conversion's overall speech quality. On the one hand, the lack of objective measures hinders choosing a training strategy and an objective function. On the other hand, most of the works in VC only report subjective scores. Some works on parallel VC report objective metrics such as the Root Mean Square Error (RMSE) [1], but those are not always correlated with human perception [2]. This is especially the case for non-parallel scenarios, where even Perceptual Evaluation of Speech Quality (PESQ), which predicts the human-perceived speech quality with respect to a target signal, does not correlate with subjective scores (see Section 5.1).

The subjective score indicates how a system works and may be biased depending on the test setup. Moreover, it is impossible to compare available systems evaluated only by different evaluators' subjective tests and under different conditions. The VC Challenge[1] circumvents this problem by providing a common evaluation dataset and performing large-scale crowd-sourced perceptual evaluations.

VC requires the factorization of speech into linguistic and non-linguistic information. Therefore, one natural approach is to use representations that lack speaker information, which are well studied in the speech recognition domain. Such representations are then concatenated with the information of the target speaker (studied in the speaker identification and verification domains) and mapped to the waveform domain yielding the VC output. One example of such an approach achieving high-quality conversions consists of first transcribing audio to text using an Automatic Speech Recognition (ASR) system and then using a Text-to-Speech (TTS) model conditioned on the target speaker and the obtained transcription [3, 4, 5].

Another common approach when dealing with non-parallel data is to train an AE model in speech reconstruction and enforce speaker independence in the latent representations. Such latent features are then used in the same fashion as the speaker-independent representations of the former approach. This method can suppress the need for annotated data and find representations that potentially preserve para-linguistical information.

In line with the VC Challenge 2020, this work uses non-parallel data for the VC model training. In particular, the proposed method is based on conditional AEs as in AutoVC [6], but with focus on fast conversions.

2. Related Works

The approaches using speaker-independent representations leverage pre-trained models for its computation. In general, these systems require large amounts of transcribed data, whose collection is very costly and time-consuming [7]. Moreover, such speaker-independent features usually lack para-linguistical information such as the intonation, which can potentially lead to conversions having different meanings in some cases. In the case of using text, the timing information is also lost, and an additional mapping to phonetic transcriptions is needed in cross-lingual settings.

Speaker independence can be explicitly enforced with an adversarial setting, where a classifier is trained to predict the speaker from the latent representation. The loss from such classifiers can then be used to learn the mapping from the input signal to the latent space [8, 9].

One can also implicitly enforce speaker independence by reconstructing the speech from the low-dimensional latent representation and uncompressed speaker information. Speaker disentanglement, in this case, follows from the redundancy principle [10]. Since the waveform generator is explicitly conditioned on the speaker identity, the feature extractor does not have to capture speaker-dependent information in the latent features. The desired speaker independence is achieved if the dimension of the latent space satisfies the following trade-off. On the one hand, it has to be sufficiently small to factor out the speaker's information. On the other hand, it has to be large enough to allow for perfect reconstruction and capture as much of the input data as possible.

AutoVC achieves a speaker-independent latent space by

[1] http://www.vc-challenge.org/

10.21437/VCC_BC.2020-21

implementing the previous approach with a simple conditional AE. The encoder network of the AE applies an information bottleneck that both reduces the number of features and downsamples the signal in the temporal dimension.

CycleVAE [11] models the information bottleneck with a Variational AutoEncoder (VAE), which means that the latent features are enforced to follow a known distribution. Vector Quantization (VQ)-VAE [12] uses VQ as an additional information bottleneck on the latent features obtained with a VAE. Time-jitter regularization, consisting of replacing each latent vector with either one or both of its neighbors, was also shown to be useful as an additional information bottleneck in top of the former approach [13]. This regularization helps to model the slowly-changing phonetic content by avoiding the use of latent vectors as individual units. The authors claimed that the latent representations found by AEs do not factor the speaker out. Instead, their findings showed that a VAE or its VQ version was required to achieve speaker disentanglement. However, [14] showed that conditional AEs with speaker-dependent encoders effectively achieve the desired speaker-independent latent representations.

3. FastVC

FastVC is an end-to-end model that performs fast many-to-many VC and is trained using non-parallel data. This system performs VC by learning a mapping between the source and converted waveforms. This latter has the same linguistic information as the source speech but different speaker information. In particular, FastVC learns this mapping with a conditional AE framework that is trained on the reconstruction of Mel-spectrograms similarly to [6]. FastVC performs VC with the three-stage model depicted in Figure 1. Its AE module is depicted in Figure 2.

Equally to AutoVC [6], FastVC uses log-scale Mel-spectrograms with 80 Mel channels as inputs. However, the Mel-spectrogram module is a Convolutional Neural Network (CNN)-based learnable module and not a fixed transformation as in AutoVC. This module can be trained if desired and is initialized to provide exact Mel-spectrograms. This allows using raw speech waveforms as input instead of Mel-spectrograms.

The theoretical guarantees justifying the VC capabilities of [6] hold under the assumption that the speaker embeddings of different utterances of the same speaker are the same, and those from different speakers are distinct. In order to satisfy this assumption, FastVC uses one-hot encoded speaker embeddings similarly to [12].

In FastVC, both the encoder $E(\cdot,\cdot)$ and the decoder $D(\cdot,\cdot)$ are conditioned on the speaker identity of the source and target speaker as in [14]. This speaker identity is concatenated with the other input signal at every time step. One of the most key design choices in implementing a conversion function for VC learned on speech reconstruction is choosing adequate information bottlenecks.

FastVC uses dimensionality reduction in the frequency dimension and temporal downsampling as in [6]. The latent features are then upsampled to match the original time rate using a causal variant of the nearest neighbor interpolation technique. This can be seen as a causal version of the time-jitter regularization proposed in [13], with the time jitter as a hyper-parameter that corresponds to the downsampling factor.

The information bottleneck introduces two pivotal hyper-parameters for the speaker disentanglement; the latent features' dimension and the downsampling factor. In particular, FastVC doubles the temporal downsampling factor with respect to [6]. This design choice achieves speaker-independent phoneme-like latent features that solve the pitch inconsistency problems of AutoVC reported in [15]. For more details on this, refer to [16].

FastVC generates speech with a sampling rate of 22050 Hz, which makes Auto-Regressive (AR) models unsuitable, especially if fast conversions are desired. [6] used WaveNet [17] conditioned on the log-scaled Mel-spectrogram as a generative model for raw speech. To achieve fast inference, FastVC resorts to using a non-AR generative model. This design choice is the main reason for the fast conversions obtained with this approach. In particular, the Mel-spectrogram inverter is chosen to be MelGAN, introduced by [18].

4. Experimental setup

4.1. Datasets

FastVC only trains on raw speech waveforms and speaker identities. This means that it does not requires any additional annotation. The main dataset used for this project is the Voice Cloning Toolkit (VCTK) described in [19]. The VCTK dataset is chosen as the main dataset for its widespread use for the VC task [12, 8, 6, 15].

The model comparison of the VC Challenge is performed with samples generated using the dataset of the same Challenge. The use of the Challenge training dataset is essential but not enough to train a model such as FastVC. In this project, the VCTK and VC Challenge datasets were simply merged, which is allowed in the Challenge.

The TIMIT dataset [20] is chosen for the latent features' analysis (see Section 5.3). This dataset is public and contains speech data and hand-verified time-aligned phoneme transcriptions. In particular, only the test partition of this dataset is used to avoid incorporating many new speakers.

4.2. Training

Generative Adversarial Networks (GANs) are notoriously difficult to train, with mode collapse and oscillations being a common problem [21]. For this reason, the basic FastVC model (denoted FastVC in Table 1) uses the pre-trained weights for MelGAN provided by [18]. The AE module in FastVC is trained from scratch to match the conditioning signal required by the Mel-spectrogram inverter. The basic version of FastVC is obtained by only training the AE module using the ADAM optimizer [22] with a learning rate of 0.001, $\beta_1 = 0.9$, and $\beta_2 = 0.99$ for 200 epochs. The training objective for this setting is presented in (2), where $\mathbf{X}$ is the input Mel-spectrogram, $\mathbf{s}$ its correspondent speaker and $\hat{\mathbf{X}}$ the AE output.

$$\mathcal{L}_{content} = \left\| E(\mathbf{X},\mathbf{s}) - E(\widehat{\mathbf{X}},\mathbf{s}) \right\|^2 \qquad (1)$$

$$\min \mathbb{E}_{\mathbf{X},\mathbf{s}} \left[\left\| \mathbf{X} - \widehat{\mathbf{X}} \right\|^2 + \| \mathbf{X} - D(E(\mathbf{X},\mathbf{s}),\mathbf{s}) \|^2 + \mathcal{L}_{content} \right] \qquad (2)$$

FastVC with end-to-end training: to allow the model to use information that may be not included in the Mel-spectrogram and generate more efficient representations for the task of VC, FastVC also allows end-to-end training. The weights obtained in the only-AE training are used as a starting point. In this setup, FastVC behaves as the generator of a GAN, which takes raw speech as input.

FastVC uses multiple discriminators that run at different rates, as proposed in [18]. To ensure that the linguistic infor-

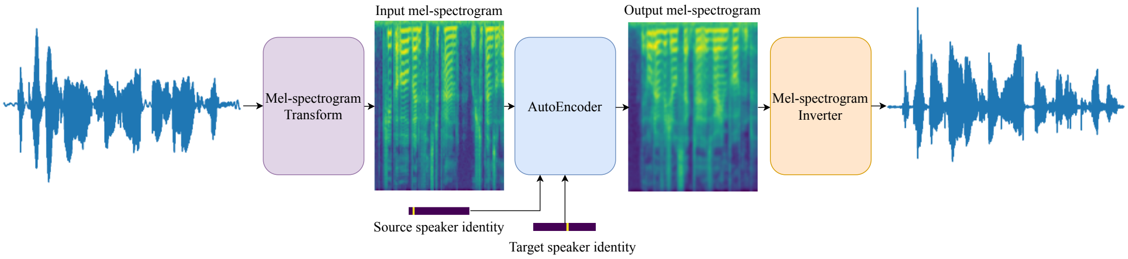

Figure 1: *FastVC model architecture during conversion mode. During training, both the the source and target speaker identities are the same.*

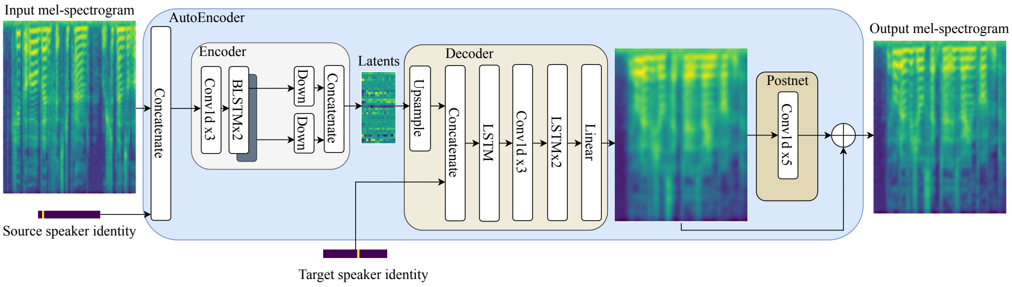

Figure 2: *Diagram of the AE module for FastVC during conversion mode. The* `AutoEncoder` *comprises the* `Encoder` *and the* `Decoder`*, but also the* `PostNet`*. The* `PostNet`*, which is proposed in [6], builds the finer details of the spectrogram, which is excessively smooth before this module.*

mation is captured, the content loss term (1) is added to the generator objective in [18]. The content loss term enforces the codes of the original and converted speech to be the same, i.e., it enforces VC to be idempotent. We speculate that this is enough to achieve quality speech that preserves the lexical content.

The content loss is weighted by a factor of 20 and added to the generator's total objective. The regularization amount is chosen to ensure that the losses have the same order of magnitude, and they both decrease individually. For this setting, ADAM is also used as the optimization algorithm. In this case, however, with a learning rate of 10^{-4}, $\beta_1 = 0.5$, and $\beta_2 = 0.9$ for 200 epochs. These specific values are suggested in [23] to train GANs with ADAM, and also used in [18].

All the experiments use a batch size of 16, and the merged dataset is randomly split into 90% for training and 10% for testing purposes. During training, FastVC is fed chunks of 8192 samples of speech sampled at 22050 Hz, while in inference, the model's input is the whole waveform.

5. Results

FastVC converts voices 4× faster than real-time, and 500× faster than AutoVC, measured on Intel(R) Core(TM) i7-8700K at 3.70GHz.

5.1. Objective assessment

One of the main difficulties in building VC models is that there are no standardized objective measures. The lack of such metrics hinders the system comparison and the performance of ablation studies. The variant of FastVC submitted to the VC Challenge was chosen based on the value given by PESQ, an objective method that rates the speech quality by predicting the Mean Opinion Score (MOS).

The fact that PESQ is not used as a standardized measure to substitute MOS is that the former requires both the desired waveform and the one generated with the evaluated system. The approach that FastVC takes to deal with non-parallel data is to learn the conversion function on the task of speech reconstruction. In this case, the PESQ measure is more suited since self-reconstruction was learned during training, and mapping to the same speaker is a valid VC instance.

Experiment	PESQ
AutoVC – baseline [6]	$\mathbf{2.56 \pm 0.23}$
FastVC with information bottleneck proposed in [6]	2.57 ± 0.25
FastVC with 10 Hz latent features	2.61 ± 0.24
FastVC with adversarial speaker classifier	2.62 ± 0.24
FastVC (VCC20 submission)	$\mathbf{2.68 \pm 0.22}$
FastVC with end-to-end training	1.56 ± 0.29
FastVC with learnable Mel-spectrogram	1.50 ± 0.40
PhonetVC (Section 5.4)	$\mathbf{1.67 \pm 0.30}$

Table 1: *Objective results performed over 100 utterances of less than 5 seconds from the test partition. The reported values are the mean and the standard deviation of the sample. First three FastVC variants are described later in Section 5.3.*

Table 1 shows the obtained results. Note that the reconstruction performance alone is not a useful metric to evaluate a VC system because it does not measure the speaker's disentanglement. In particular, perfect reconstruction can be achieved if there is no information bottleneck and thus the latent features are not speaker-independent. Therefore, this metric should be used for systems with speaker-independent inputs.

FastVC with end-to-end training performs worse in terms of PESQ than FastVC. This can be justified because, in the end-to-end training, the aim is not to match the input Mel-spectrogram but to maximize the GAN objective. A future subjective evaluation would be needed to confirm if the PESQ also correlates with the perceived quality in such cases.

The use of this metric with parallel utterances aligned using Dynamic Time Warping (DTW) was also explored. However, in this case, the results were inconclusive and not related at all with perceptual scores. The ill-posedness of the problem can justify this; a sound output other than the time-aligned parallel utterance (or the input utterance in the evaluation of reconstruction, especially on the end-to-end case) may be obtained.

5.2. Subjective assessment

The subjective scores for the cross-lingual VC task are presented in the VC Challenge 2020 paper. FastVC is represented with the label **T15**. You can also compare the VC Challenge baselines, AutoVC, and the proposed FastVC at `https://barbany.github.io/fast-vc/`.

5.3. Latent space analysis

5.3.1. Speaker independence

Prosodic information leaks through the bottleneck of AutoVC, causing the target pitch to fluctuate unnaturally [15]. To tackle this issue, the authors proposed in [6] to remove the speaker identity from the latent representations and the prosodic information. The temporal downsampling factor proposed in [15] matches the design choice of FastVC. With this value, FastVC outputs do not have the unnatural pitch jumps of AutoVC without the need of disentangling the prosodic information from the latent features and introducing the synthetic target prosody. Refer to [16] for more details.

FastVC requires that the latent features are speaker-independent, but this is not explicitly enforced. The fact that the encoder disentangles the speaker in an unsupervised fashion can be explained with the redundancy principle [10]. However, adversarial training of the latent representations as in [8] could further disentangle the speaker's information and downplay the information bottleneck's design choices.

To confirm if the redundancy principle suffices, a variant of FastVC with an adversarial speaker classifier was implemented. In particular, an adaptation of the discriminator used to achieve class-independent latent representations in [24] is implemented. The minimax game here is for the encoder to seek class-independent latent features and the classifier to classify them correctly. The classifier is trained with the cross-entropy loss on the speaker labels using the ADAM optimizer with a learning rate of 0.001, $\beta_1 = 0.9$, and $\beta_2 = 0.99$. The negative loss, termed as domain confusion loss in [24], is added as a regularizer to (2) with a weighting of 0.1 so that each individual objective had the same order of magnitude.

Even if the classifier network was trained simultaneously as FastVC, the prediction accuracy was 0% when the speaker-independence signal was used with the latent features and when it was not. These results were obtained with a model trained using the 278 speakers resulting from the mix of the VCTK corpus and the test partition of the TIMIT dataset. The speaker-independence results suggest that the redundancy principle is enough to achieve speaker-independence, which is in line with the results reported in [6, 15].

5.3.2. Phonetic similarity

Similarly to [12, 13], the latent features of FastVC lack speaker information and are potentially similar to phonemes. A perceptron is used to find a hypothetically simple correspondence between phonemes and the latent features. This model is trained using the latent representations extracted from the TIMIT test data with a trained FastVC network. The obtained latent features are randomly split into the train (70%), validation (10%), and test (20%) sets.

The information bottleneck on the temporal dimension of FastVC yields a latent representation with a 2.5 Hz rate. This rate is a factor of 10 lower than the rate of the latent features in [12]. The average phoneme rate is around 10 Hz [25, 26], which means that each latent vector at a given time represents more than one phoneme. For the classification task, each latent vector was assumed to represent the phoneme with a larger intersection in the temporal domain.

The phoneme classifier was trained by minimizing the cross-entropy loss with Stochastic Gradient Descent (SGD) and early stopping on the loss on the validation partition of the dataset containing the latent features from the TIMIT test data. This classifier correctly classified 42.45% of the latent features. In contrast, a random classifier and a classifier always choosing the prior most likely phoneme on the train partition had an accuracy of 2.44% and 9.43%. These results suggest that there is indeed a correspondence between the latent features and phonemes. For comparison, VQ-VAE [12] uses a 128-dimensional discrete space and obtains a classification accuracy of 49.3%, while choosing the prior most likely phoneme gives a 7.2%. A classification drop from the results in [12] is expected due to the lower rate representation and the classifier's simplicity.

Even if the latent representations' low rate suggested that a latent vector represents a combination of sounds rather than a single phoneme, the number of distinct units with groups of phonemes exponentially grows with the group size. This growth implies that there are more classes to predict, and some may not even be seen during training.

5.4. PhonetVC

PhonetVC is a variant of the proposed model designed to confirm the benefits of using the latent features obtained by FastVC instead of speaker-independent speech features. PhonetVC uses an estimation of the Phonological Log-Likelihood Ratio (PLLR) features computed with Phonet [27] instead of the latent representations obtained by the encoder network in Figure 2. The resulting model works with speech at 16 kHz, and the Decoder, Postnet, and Mel-spectrogram inverter are jointly trained from scratch using the MelGAN training objective [18].

6. Conclusions

This work proposed a fast and competitive VC system. It is worse at capturing the speaker's style of speakers with little data in comparison to its quality performance (see subjective results in the VC Challenge paper). This is justified by the fact that the training dataset is very imbalanced concerning the language, and the performance could be degraded for non-English speakers. One possible approach to tackle the language imbalance problem is to incorporate additional non-English speech datasets to balance the languages. However, the percentage of data per speaker on the VC Challenge would be even smaller in this case. A different approach to tackle dataset imbalance is the multi-reader technique described in [28].

7. Acknowledgements

We thank Kaizhi Qian for providing the non open-sourced full code of AutoVC used as a starting point for this project.

8. References

[1] H. Du, X. Tian, L. Xie, and H. Li, "Effective Wavenet Adaptation for Voice Conversion with Limited Data," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7779–7783.

[2] B. Sisman, J. Yamagishi, S. King, and H. Li, "An Overview of Voice Conversion and its Challenges: From Statistical Modeling to Deep Learning," 2020.

[3] T. Hayashi, R. Yamamoto, K. Inoue, T. Yoshimura, S. Watanabe, T. Toda, K. Takeda, Y. Zhang, and X. Tan, "ESPnet-TTS: Unified, Reproducible, and Integratable Open Source End-to-End Text-to-Speech Toolkit," 2019.

[4] H. Inaguma, S. Kiyono, K. Duh, S. Karita, N. E. Y. Soplin, T. Hayashi, and S. Watanabe, "ESPnet-ST: All-in-One Speech Translation Toolkit," *arXiv preprint arXiv:2004.10234*, 2020.

[5] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. Enrique Yalta Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, "ESPnet: End-to-End Speech Processing Toolkit," in *Interspeech*, 2018, pp. 2207–2211. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1456

[6] K. Qian, Y. Zhang, S. Chang, X. Yang, and M. Hasegawa-Johnson, "AUTOVC: Zero-Shot Voice Style Transfer with Only Autoencoder Loss," 2019.

[7] S. Pascual, M. Ravanelli, J. Serrà, A. Bonafonte, and Y. Bengio, "Learning Problem-Agnostic Speech Representations from Multiple Self-Supervised Tasks," in *Proc. of the Conf. of the Int. Speech Communication Association (INTERSPEECH)*, 2019, pp. 161–165. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-2605

[8] J. Chou, C. Yeh, H. Lee, and L. shan Lee, "Multi-target Voice Conversion without Parallel Data by Adversarially Learning Disentangled Audio Representations," 2018.

[9] H. Kameoka, T. Kaneko, K. Tanaka, and N. Hojo, "StarGAN-VC: Non-parallel many-to-many voice conversion with star generative adversarial networks," 2018.

[10] H. Barlow, "Unsupervised learning," *Neural Computation*, vol. 1, no. 3, pp. 295–311, 1989. [Online]. Available: https://doi.org/10.1162/neco.1989.1.3.295

[11] P. L. Tobing, Y.-C. Wu, T. Hayashi, K. Kobayashi, and T. Toda, "Non-Parallel Voice Conversion with Cyclic Variational Autoencoder," 2019.

[12] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural Discrete Representation Learning," *CoRR*, vol. abs/1711.00937, 2017. [Online]. Available: http://arxiv.org/abs/1711.00937

[13] J. Chorowski, R. J. Weiss, S. Bengio, and A. van den Oord, "Unsupervised speech representation learning using WaveNet autoencoders," *CoRR*, vol. abs/1901.08810, 2019. [Online]. Available: http://arxiv.org/abs/1901.08810

[14] M. Heck, S. Sakti, and S. Nakamura, "Unsupervised Linear Discriminant Analysis for Supporting DPGMM Clustering in the Zero Resource Scenario," *Procedia Computer Science*, vol. 81, pp. 73–79, 12 2016.

[15] K. Qian, Z. Jin, M. Hasegawa-Johnson, and G. J. Mysore, "F0-Consistent Many-To-Many Non-Parallel Voice Conversion Via Conditional Autoencoder," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2020. [Online]. Available: http://dx.doi.org/10.1109/ICASSP40776.2020.9054734

[16] O. Barbany, "Speech Style Transfer," Master's thesis, École Polytechnique Fédérale de Lausanne (EPFL), 2020. [Online]. Available: https://infoscience.epfl.ch/record/279866?&ln=en

[17] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, "WaveNet: A Generative Model for Raw Audio," *CoRR*, vol. abs/1609.03499, 2016. [Online]. Available: http://arxiv.org/abs/1609.03499

[18] K. Kumar, R. Kumar, T. de Boissiere, L. Gestin, W. Z. Teoh, J. Sotelo, A. de Brébisson, Y. Bengio, and A. C. Courville, "MelGAN: Generative Adversarial Networks for Conditional Waveform Synthesis," in *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 14 910–14 921. [Online]. Available: http://papers.nips.cc/paper/9629-melgan-generative-adversarial-networks-for-conditional-waveform-synthesis.pdf

[19] C. Veaux, J. Yamagishi, and K. MacDonald, "CSTR VCTK Corpus: English Multi-speaker Corpus for CSTR Voice Cloning Toolkit," University of Edinburgh. The Centre for Speech Technology Research (CSTR), 2016.

[20] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, N. L. Dahlgren, and V. Zue, "TIMIT Acoustic-Phonetic Continuous Speech Corpus LDC93S1," Philadelphia: Linguistic Data Consortium, 1993.

[21] K. J. Liang, C. Li, G. Wang, and L. Carin, "Generative Adversarial Network Training is a Continual Learning Problem," 2018.

[22] D. P. Kingma and J. Ba, "ADAM: A Method for Stochastic Optimization," 2014.

[23] C. Daskalakis, A. Ilyas, V. Syrgkanis, and H. Zeng, "Training GANs with Optimism," *CoRR*, vol. abs/1711.00141, 2017. [Online]. Available: http://arxiv.org/abs/1711.00141

[24] N. Mor, L. Wolf, A. Polyak, and Y. Taigman, "A Universal Music Translation Network," in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=HJGkisCcKm

[25] S. Van Kuyk, W. B. Kleijn, and R. C. Hendriks, "On the information rate of speech communication," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5625–5629.

[26] M. Cernak, A. Asaei, and A. Hyafil, "Cognitive speech coding examining the impact of cognitive speech processing on speech compression," *IEEE Signal Processing Magazine*, vol. 35, no. 3, pp. 97–109, 2017. [Online]. Available: http://infoscience.epfl.ch/record/231842

[27] J. C. Vásquez-Correa, P. Klumpp, J. R. Orozco-Arroyave, and E. Nöth, "Phonet: A Tool Based on Gated Recurrent Neural Networks to Extract Phonological Posteriors from Speech," in *Proc. Interspeech 2019*, 2019, pp. 549–553. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-1405

[28] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized End-to-End Loss for Speaker Verification," 2017.