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

The NeteaseGames System for Voice Conversion Challenge 2020 with Vector-quantization Variational Autoencoder and WaveNet

Haitong Zhang

NetEase Games AI Lab
zhanghaitong01@corp.netease.com

Abstract

This paper presents the description of our submitted system for Voice Conversion Challenge (VCC) 2020 with vector-quantization variational autoencoder (VQ-VAE) with WaveNet as the decoder, i.e., VQ-VAE-WaveNet. VQ-VAE-WaveNet is a nonparallel VAE-based voice conversion that reconstruct the acoustic features along with separating the linguistic information with speaker identity. The model is further improved with the WaveNet cycle as the decoder to generate the high-quality speech waveform, since WaveNet, as an autoregressive neural vocoder, has achieved the STOA result of waveform generation. In practice, our system can be developed with VCC 2020 dataset for both Task 1 (intra-lingual) and Task 2 (cross-lingual). However, we only submitted our system for the intra-lingual voice conversion task. The results of VCC 2020 have demonstrated that our system VQ-VAE-WaveNet achieves: 3.04 mean opinion score (MOS) in naturalness and 3.28 average score in similarity (speaker similarity percentage (Sim) of 75.99%) for Task 1. What's more, our system performs well in some objective evaluations. Specifically our system achieved an average score of 3.95 in naturalness in automatic naturalness prediction and ranked the 6th and 8th, respectively in ASV-based speaker similarity and spoofing countermeasures.

Index Terms: Voice Conversion Challenge, VQ-VAE, WaveNet

1. Introduction

Voice conversion (VC) aims to changing the speaker identity of a source utterance into that of a desired target speaker while retaining the linguistic contents of the source utterance [1]. VC is very useful for various new applications, such as expressive speech synthesis [2], impaired speech improvement [3].

Voice conversion challenge (VCC) [1] aims to better understand different VC techniques built on a common released dataset. During the past six years, there VC challenges have been carried out, i.e., VCC 2016 [4], VCC 2018 [5], and VCC 2020 [6], with a common goal, but a slight different focus. In VCC 2020, two new tasks are considered. The first one is semi-parallel intra-lingual VC, where the training data comes from one language and small part of them are parallel. The second task is cross-lingual VC, where the source speaker is different from the target one both in terms of language and content.

Since the challenge focus on non-parallel VC, it should be noted that a promising paradigm for non-parallel VC is a recognition-synthesis framework [7, 8]. The idea is to first extract the linguistic content from the source speech and concatenated with the speaker identity/feature to generate the speech in target voice. A popular type of models are based on phonetic posteriorgram (PPG). In this type of models, the source speech is usually fed into an automatic speech recogntion (ASR) model to extract the PPG, then a synthesis model is applied to synthesize the speech in target speaker's voice by conditioning on the PPG. Another group of VC techniques for non-parallel VC are based on autoencoder-like model [9–11]. These models are developed to disentangle the speaker characteristics from the linguistic contexts by using only reconstruction on the spectral feature level.

Both types of techniques mentioned have been proved efficient in non-parallel VC. However, considering the first type of techniques reply much on the performance ASR model (which also requires much more work on data annotation/labelling), we focus on using auto-encoder-like model to perform non-parallel VC for Task 1.

Specifically, we built our system based on VQ-VAE model [12]. We used VQ-VAE mainly because the vector-quantization part can discretize the continuous representations into a fixed-number of embeddings, and these embeddings are proven to be phoneme-like in [13, 14]. Thus, we believe VQ-VAE would perform better in extracting the linguistic context.

Many great VC systems consist of a conversion model or a neural vocoder [15, 16], with the conversion model to convert the source acoustic features into target speaker's voice, and the vocoder to transform the converted features into speech waveform. These two models are usually trained in a separate mode. As a result, there would be some mismatches between the the data distribution of the converted features and that of the ground truth features used to train the vocoder. Usually these mismatches could be reduced by training or fine-tuning the neural vocoder by using the converted features and/or the natural features. This can be regarded as a method of data augmentation to regularize the neural vocoder. Based on this observation, we combined the vocoder part into the conversion model in order to train the models jointly, which can perform voice conversion and produce the speech waveform directly.

The rest of this paper is organized as follows: Section 2 details our system. Section 3 provides the evaluation results and our system performance. Conclusions are given in Section 4.

2. NeteaseGames System

In this section, we describes the overall structure of our submitted system for Task 1 of VCC 2020, as illustrated in Figure 1. In details, our system mainly consists of three parts, namely the encoder, the vector-quantization part, WaveNet-baed decoder. We describe each module as follows.

2.1. Encoder

The encoder part is partly composed of stacks of convolutional layers and fully-connected layers, and residual connections. The encoder takes the source features $X = x_1, x_2, ..., x_T$ and

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

10.21437/VCC_BC.2020-27

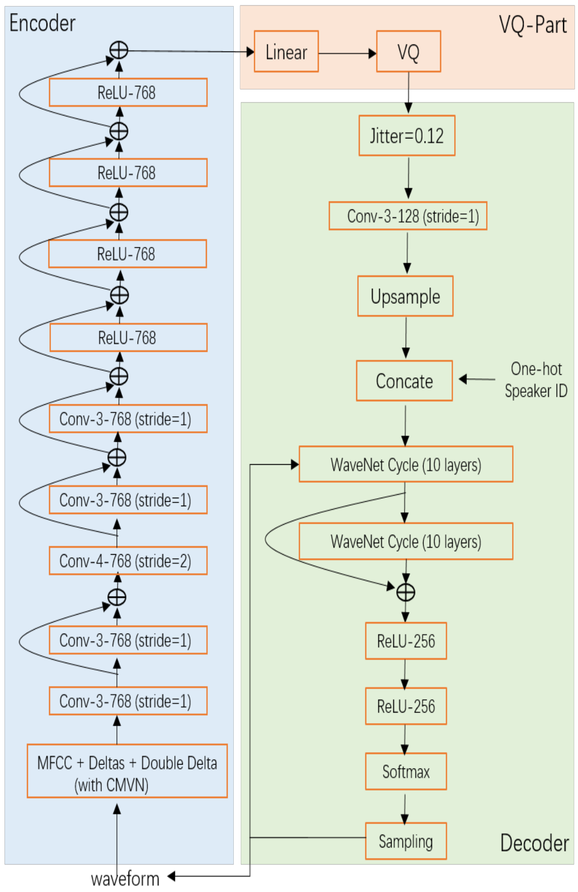

Figure 1: *The model structure of our VC system.*

learn the high-level representation through non-linear transformation. The whole procedure is formulated as below.

$$\begin{aligned}
h_1 &= Conv_{3*768}(X); \\
h_2 &= Conv_{3*768}(h_1) + h_1; \\
h_3 &= Conv_{4*768}(h_2), stride = 2 \\
h_4 &= Conv_{3*768}(h_3) + h_3; \\
h_5 &= Conv_{3*768}(h_4) + h_4; \\
h_6 &= ReLU(wh_5 + b) + h_5; \\
h_7 &= ReLU(wh_6 + b) + h_6; \\
h_8 &= ReLU(wh_7 + b) + h_7; \\
h_9 &= ReLU(wh_8 + b) + h_8
\end{aligned} \tag{1}$$

where $3 * 768$ means the kernel size is 3 and filter size is 768, $+$ means residual connection, w and b is the weight matrix and bias. Downsampling effect is achieved by setting stride step into 2 in the third convolutional layer.

2.2. Vector quantization Part

Vector quantization (VQ) part is an important part in VQ-VAE, used to quantization the continuous hidden representation into fixed number of embeddings, as illustrated in Figure 2.

A codebook dictionary $e = C * D$ is pre-defined , where C is the number of latent embeddings in the dictionary and D is the dimension of each embedding. The VQ part takes the

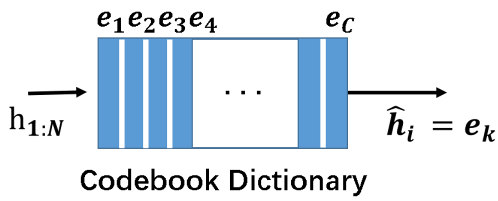

Figure 2: *The VQ part in VQ-VAE.*

encoded representation $h_9 = h_{1:N}$, where N depends on the length T and the number of down-sampling layers in the encoder. Then the continuous latent representations $h_{1:N}$ can be mapped into $\hat{h}_{1:N}$ by finding the nearest pre-defined discretized embedding in the dictionary as $\hat{h} = e_k$, where $k = argmin_j$ $||h - e_j||$, and e_j is the j-th embedding in the codebook dictionary, and $j \in 1, 2, \cdots, C$.

2.3. Jitter Mechanism

Jitter mechanism is inspired by [13]. We insert the jitter mechanism after the VQ part in order to enforce some consecutive frames would be quantized into the same embedding in the codebook doctionary, since the speech signal changes slowly. The jitter mechanism replace the embedding at current frame with that at the last frame and at the next frame at a certain probability.

2.4. Decoder

After jitter mechanism, the learned embedding sequence is further processed by one convolutional layer before upsampling layer. Unsampling is performed to match the length of the hidden representation sequence and the input feature sequence length. After upsampling, the sequence is concatenated with one-hot speaker embedding at each time index to form the final input of the WaveNet-like decoder.

Conventional vocoders based on source-filter theory severely limit the synthesis quality [17, 18] . The neural-network-based vocoders greatly improve the quality of synthesized speech by directly modeling the speech waveform and by-passing the traditional speech vocoders. WaveNet based neural vocoder, especially, could generate the waveform whose quality is comparable to that of the natural speech. WaveNet is an autoregressive generative model that models waveform directly. The original WaveNet is realized by the use of conditional WaveNet, in which acoustic feature is used as the conditional input to generate waveform (see Figure 3). Given a sequence of waveform $X = x1, x2, ..., xt$, the joint probability of all these samples is represented as follows:

$$p(x|h;\theta) = \sum_{t=1}^{T} p(x_t|x_1, x_2, ..., x_t, h; \theta) \tag{2}$$

where h is the local conditional features, θ represents the parameters of the WaveNet part. $p(x_t|x_1, x_2, ..., x_{t-1})$ denotes a conditional dependence between current waveform samples and the previous samples in a long range. In original WaveNet model, the local conditional features are ground truth acoustic features, in our model, these local conditional features are the learned features given the ground truth features through the mentioned nonlinear transformation.

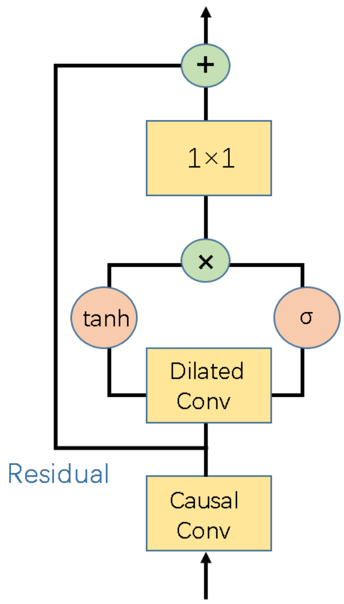

Figure 3: *The main structure in WaveNet-like decoder.*

As the WaveNet architecture in [15], we adopted 20 dilated convolution layers, grouped into 2 dilation cycles, i.e., the dilation rate of layer $k(k = 0, 1, ..., 19)$ is $2^{(k(mod10))}$. The filter width is 2, and residual and skip channels are set into 256, and the softmax output dimension is 128. The model is optimized with Adam algorithm.

2.5. Training steup

The training data released for Task 1 of Voice Conversion Challenge 2020 consists of 8 speakers in English language. That of source speakers consist of 2 males and 2 females. The set of target speakers includes another 4 English speaker (2 males and 2 females). The number of utterances for each speaker is 70 with 20 parallel utterances between source and target speakers. To further improve the model robustness to unseen data, we augmented the training data with speech of 100 speakers from VCTK speech corpus [19]. The model was trained with a mixture of training data released and VCTK data, then fine-tuned with the training data released only. We kept five utterances from each speaker for validation purpose. We did not perform any data labelling and checking.

We used MFCC as model inputs instead of waveform, since we believe using MFCC is beneficial to disentangling speaker characteristics from the input features. Specifically, we used first 13 MFCC features. We then concatenate MFCC with delta and double delta features and then apply CMVN across each speaker. We set the jitter probability into 0.12 by empirical experiments. The whole model is trained using Adam algorithm, with a starting learning rate of $2.5e-4$ and halving into $1e-5$ until convergence.

3. Evaluations

Three baseline systems are also included in the task. The first baseline system was built using PPG-based VC techniques, which achieved the best performance in VCC 2018 [8], but not publicly available. The second baseline is a cascaded ASR and TTS system using sequence-to-sequence models [20], whose open implementation is freely available at `https://github.com/espnet/espnet/tree/master/egs/vcc20`. The third baseline is CycleVAEPWG, where CycleVAE is used for voice conversion and Parallel WaveGAN (PWG) for waveform generation [21], and the implementation has been made freely available at `https://github.com/bigpon/vcc20_baseline_cyclevae`. Including three baselines, there were 31 participant systems submitted to Task 1.

The official evaluation test consists of 25 unseen utterances in English. In the task one, each source utterances are required to be converted into four target speakers' voice. Thus, the whole set of submitted utterances for the task includes in total 400 utterances. But only 5 utterances (E30001, E30002, E30003, E30004, E30005) are selected for perceptual evaluation. A large-scale listening test was conducted for subjective evaluation, where there were 206 Japanese listeners and 68 English listeners.

3.1. Naturalness Test

Figure 4 shows the scatter plot matching naturalness of Japanese listeners and English ones. Three baselines are denoted as T11, T22, and T16, respectively. TAR and SOU refer to the natural speech of the target speaker and the source speaker, respectively. From the plot, we can witness that there four systems perform better than T11, the best performing system in VCC 2018, and the difference between T10 or T13 and T11 is significant. It indicates that VC performance has improved within the past two years in terms of naturalness. Another important highlight is that most of the top performing systems are based on PPG or ASR or leveraged with TTS, which indicates that more labelled data is helpful for VC performance. Our system, T04, achieved an average MOS score of 3.04 (the average score of Japanese listeners and English ones). It should be noted that another two systems based on VQ-VAE achieved an average score of 2.44 and 3.16. Although the PPG or recognition-based models achieve the best performances, our systems, together with T20, achieved the best performance in naturalness as no supervised learning (such as ASR or TTS) is involved.

3.2. Similarity Test

Figure 5 shows the scatter plot matching similarity of Japanese listeners and English ones. It is shown that our system achieved an average score of 3.28 in similarity. What is interesting is that system T20 achieved an average score of 2.89. This difference could be attributed to the joint training strategy of our system. Our system also perform the best as no unsupervised learning is considered.

3.3. Objective evaluation

Besides the subjective evaluation results, objective evaluations were also conducted [22]. The evaluations include (1) text-independent ASV for speaker similarity, text-independent CM for real-vs-fake assessment, automatic MOS prediction for quality, and ASR for intelligibility, as shown in Table 1.

Three kinds of ASV errors are included in the assessment, namely ASV EER, P_{fa}^{tar}, and P_{miss}^{src}. Overall, our system T04 ranked the 6th, achieving 45.13% , 99% , and 100% respectively for these three criteria.

The spoofing countermeasures were also included. The spoofing countermeasures is useful to defending various kinds

Table 1: *Objective assessments on VCC 2020*

Measurement Tool	Type of Measure	Metric
Automatic Speaker Verification	Target-Spoof Assessment	Equal Error Rate
Automatic Speaker Verification	Converted Source $\Longleftrightarrow$ $TargetSimilarity$	Pfa(tar)
Automatic Speaker Verification	Converted Source $\Longleftrightarrow$ $SourceSimilarity$	Pmiss(src)
Spoofing Countermeasure	Quality	Mean Opinion Score
Objective Mean Opinion Score	Intelligibility	Equal Error Rate
Automatic Speech Recognition	Real-Fake Assessment	Word Error Rate

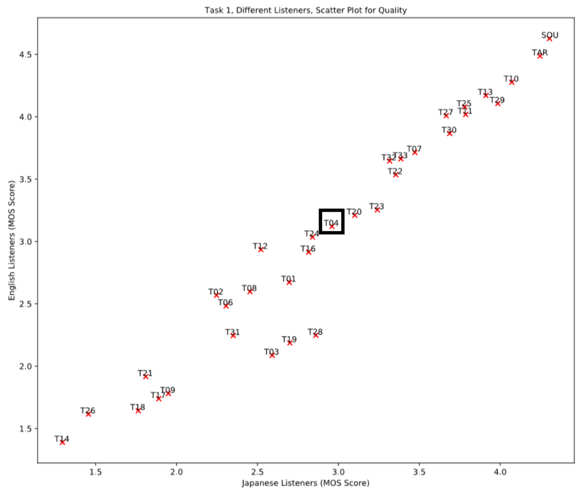

Figure 4: *Scatter plot matching naturalness of Japanese listeners and English listeners for Task 1.*

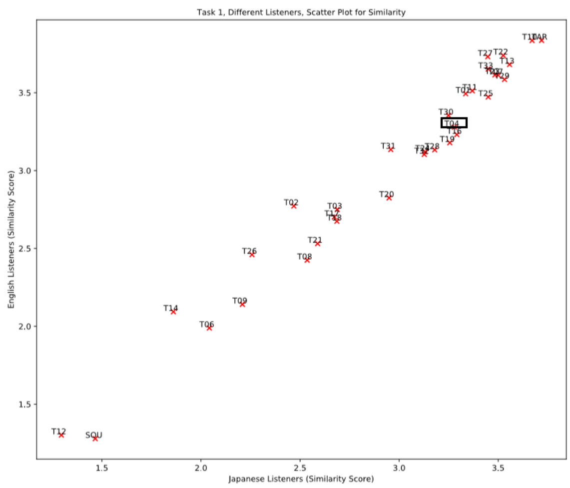

Figure 5: *Scatter plot matching similarity of Japanese listeners and English listeners for Task 1.*

of attacks to ASV systems. Our system ranked the 8th in all submitted systems.

An automatic predicted MOS [23] was also reported. Our system achieved the 3rd place among all systems, with an average score of 3.92. But the predicted MOS decreases into 3.25 as a different training dataset is used for the prediction model. At last, our system achieved a WER of about 20%.

4. Conclusion

This paper presents the details of our submitted system and summarizes the results in VCC 2020. We built our system based on VQ-VAE model. To further improve the performance we replaced the decoder in VQ-VAE with WaveNet-like decoder to directly produce speech waveform. As a result, the whole model can be trained jointly. Jitter mechanism was also applied to improve disentangling the linguistic content and speaker characteristics by our preliminary experiments. The official results show that our model achieved a naturalness and similarity score above 3. And our system is the best performing one as far as no supervised learning is involved. What is interesting is that our system achieved the third place in automatic naturalness prediction. What's more, our model performed well in ASV based speaker similarity assessments and spoofing countermeasures, ranking the 6th and 8th, respectively.

Although our system achieved reasonably good results in the task, there is still a gap between our system performance and the top-performance systems based on recognition or PPG or TTS. It shows that incorporating more labelled data or supervised learning results in a better conversion performance.

5. References

[1] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," *Journal of the Acoustical Society of Japan (E)*, vol. 11, no. 2, pp. 71–76, 1990.

[2] A. Kain and M. W. Macon, "Spectral voice conversion for text-to-speech synthesis," in *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181)*, vol. 1. IEEE, 1998, pp. 285–288.

[3] K. Tanaka, T. Toda, G. Neubig, S. Sakti, and S. Nakamura, "A hybrid approach to electrolaryngeal speech enhancement based on spectral subtraction and statistical voice conversion." in *INTERSPEECH*, 2013, pp. 3067–3071.

[4] T. Toda, L.-H. Chen, D. Saito, F. Villavicencio, M. Wester, Z. Wu, and J. Yamagishi, "The voice conversion challenge 2016." in *Interspeech*, 2016, pp. 1632–1636.

[5] J. Lorenzo-Trueba, J. Yamagishi, T. Toda, D. Saito, F. Villavicencio, T. Kinnunen, and Z. Ling, "The voice conversion challenge 2018: Promoting development of parallel and nonparallel methods," *arXiv preprint arXiv:1804.04262*, 2018.

[6] Y. Zhao, W.-C. Huang, X. Tian, J. Yamagishi, R. K. Das, T. Kinnunen, Z. Ling, and T. Toda, "Voice conversion challenge 2020 — intra-lingual semi-parallel and cross-lingual voice conversion —," in *ISCA Joint Workshop for the Blizzard Challenge and Voice Conversion Challenge 2020*. ISCA, 2020, pp. XXX–XXX.

[7] L. Sun, K. Li, H. Wang, S. Kang, and H. Meng, "Phonetic posteriorgrams for many-to-one voice conversion without parallel data training," in *2016 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2016, pp. 1–6.

[8] L.-J. Liu, Z.-H. Ling, Y. Jiang, M. Zhou, and L.-R. Dai, "Wavenet vocoder with limited training data for voice conversion." in *Interspeech*, 2018, pp. 1983–1987.

[9] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H.-M. 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)*. IEEE, 2016, pp. 1–6.

[10] H. Kameoka, T. Kaneko, K. Tanaka, and N. Hojo, "Acvae-vc: Non-parallel many-to-many voice conversion with auxiliary classifier variational autoencoder," *arXiv preprint arXiv:1808.05092*, 2018.

[11] K. Qian, Y. Zhang, S. Chang, X. Yang, and M. Hasegawa-Johnson, "Autovc: Zero-shot voice style transfer with only autoencoder loss," *arXiv preprint arXiv:1905.05879*, 2019.

[12] A. Van Den Oord, O. Vinyals *et al.*, "Neural discrete representation learning," in *Advances in Neural Information Processing Systems*, 2017, pp. 6306–6315.

[13] J. Chorowski, R. J. Weiss, S. Bengio, and A. van den Oord, "Unsupervised speech representation learning using wavenet autoencoders," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 27, no. 12, pp. 2041–2053, 2019.

[14] H. Zhang and Y. Lin, "Unsupervised learning for sequence-to-sequence text-to-speech for low-resource languages," *arXiv preprint arXiv:2008.04549*, 2020.

[15] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[16] N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimberg, A. v. d. Oord, S. Dieleman, and K. Kavukcuoglu, "Efficient neural audio synthesis," *arXiv preprint arXiv:1802.08435*, 2018.

[17] M. Morise, F. Yokomori, and K. Ozawa, "World: a vocoder-based high-quality speech synthesis system for real-time applications," *IEICE TRANSACTIONS on Information and Systems*, vol. 99, no. 7, pp. 1877–1884, 2016.

[18] H. Kawahara, "Straight, exploitation of the other aspect of vocoder: Perceptually isomorphic decomposition of speech sounds," *Acoustical science and technology*, vol. 27, no. 6, pp. 349–353, 2006.

[19] C. Veaux, J. Yamagishi, K. MacDonald *et al.*, "Superseded-cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit," 2016.

[20] W.-C. Huang, T. Hayashi, S. Watanabe, and T. Toda, "The sequence-to-sequence baseline for the voice conversion challenge 2020: Cascading asr and tts," in *ISCA Joint Workshop for the Blizzard Challenge and Voice Conversion Challenge 2020*. ISCA, 2020, pp. XXX–XXX.

[21] P. L. Tobing, Y.-C. Wu, and T. Toda, "The baseline system of voice conversion challenge 2020 with cyclic variational autoencoder and parallel wavegan."

[22] R. K. Das, T. Kinnunen, W.-C. Huang, Z. Ling, J. Yamagishi, Y. Zhao, X. Tian, and T. Toda, "Predictions of subjective ratings and spoofing assessments of voice conversion challenge 2020 submissions," in *ISCA Joint Workshop for the Blizzard Challenge and Voice Conversion Challenge 2020*. ISCA, 2020, pp. XXX–XXX.

[23] C.-C. Lo, S.-W. Fu, W.-C. Huang, X. Wang, J. Yamagishi, Y. Tsao, and H.-M. Wang, "Mosnet: Deep learning based objective assessment for voice conversion," *arXiv preprint arXiv:1904.08352*, 2019.