INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

S2VC: A Framework for Any-to-Any Voice Conversion with Self-Supervised Pretrained Representations

Jheng-hao Lin[1]*, Yist Y. Lin*[2]*, Chung-Ming Chien*[3]*, Hung-yi Lee*[4]

College of Electrical Engineering and Computer Science, National Taiwan University, Taiwan
{[1]r08922049, [2]r08922048, [3]r08922080, [4]hungyilee}@ntu.edu.tw

Abstract

Any-to-any voice conversion (VC) aims to convert the timbre of utterances from and to any speakers seen or unseen during training. Various any-to-any VC approaches have been proposed like AUTOVC, AdaINVC, and FragmentVC. AUTOVC, and AdaINVC utilize source and target encoders to disentangle the content and speaker information of the features. FragmentVC utilizes two encoders to encode source and target information and adopts cross attention to align the source and target features with similar phonetic content. Moreover, pretrained features are adopted. AUTOVC used d-vector to extract speaker information, and self-supervised learning (SSL) features like wav2vec 2.0 is used in FragmentVC to extract the phonetic content information. Different from previous works, we proposed S2VC that utilizes Self-Supervised features as both source and target features for the VC model. Supervised phoneme posteriorgram (PPG), which is believed to be speaker-independent and widely used in VC to extract content information, is chosen as a strong baseline for SSL features. The objective evaluation and subjective evaluation both show models taking SSL feature CPC as both source and target features outperforms that taking PPG as source feature, suggesting that SSL features have great potential in improving VC.

Index Terms: voice conversion, self-supervised learning, representation learning, any-to-any

1. Introduction

Self-supervised learning (SSL) [1, 2] has obtained impressive results these years in different domains, including computer vision, natural language processing, and speech processing. The self-supervised training regime does not rely on human annotations of the data, which is expensive to collect and thus benefits from the use of a large amount of unlabeled data. SSL models pretrained on speech corpora have been shown to be able to extract speech representations that can be used in downstream tasks such as automatic speech recognition, speaker recognition, and speech translation [3, 4, 5].

VC aims to convert a source utterance to sound like spoken by a target speaker while preserving the original phonetic content. The conversion can be achieved by disentangling the content and speaker information from the source and target utterances, respectively, then combining them and synthesizing the converted utterance. Supervised pretrained representations have long been used to provide content or speaker information for VC tasks. Phoneme posteriorgram (PPG) especially is very popular among VC implementations [6, 7, 8]. PPG is speaker-independent and suitable for removing the speaker characteristics from the voice to be converted. The speaker representations pretrained by speaker recognition tasks such as d-vector [9] and x-vector [10] have been widely used to provide speaker information.

On the other hand, SSL representations can be utilized for phoneme recognition and speaker classification [3, 4, 5], which indicates that both phonetic and speaker information are hereditary contained in SSL representations and thus make them possible to be used in the voice conversion (VC) task. Several previous works have tried to introduce SSL representations to the VC task. For example, Huang et al. [11] proposed a sequence-to-sequence VC framework where SSL representations were used to capture the phonetic information of the source utterance. However, their approach can only convert the utterance to a predefined target speaker. On the other hand, FragmentVC [12] is an any-to-any VC model which can convert the speech of an arbitrary source speaker to any target speaker, even speakers unseen during the training time. It also used pretrained SSL models to extract the content information from source utterances.

In this paper, we aim to improve any-to-any VC (also called one-shot VC) [13, 14, 15, 16], which is one of the most difficult VC settings, by involving pretrained SSL representations. Different from previous works [11, 12], we extract not only the phonetic information but also the target speaker information from the SSL representations. Several different SSL models, including Autoregressive Predictive Coding (APC) [3], Contrastive Predictive Coding (CPC) [17], and wav2vec 2.0 [18], are investigated, and we also compare the performance of these SSL representations with supervised representations such as PPG representation [1]. The results show that SSL representations achieve comparable and even better performance than PPG representation on both subjective and objective evaluation.

2. Methods

The overall framework of S2VC is in Fig. 1. As shown in the figure, we adopt pretrained SSL models to extract source and target features. In the following subsections, we first introduce the foundation of this work: FragmentVC [12], an architecture potential to utilize any kinds of speech representations. Followed by the modification we made to FragmentVC to further improve the performance. Then briefly describe several SSL features tested in our framework.

2.1. Baseline: FragmentVC

The overall framework evolves from FragmentVC. FragmentVC is a deep exemplar-based model consisting of a source encoder, a target encoder, cross attention modules, and a decoder. The conceptual illustration of cross attention is in Fig. 2, which takes one output feature from the source encoder (Q) and two output features from the target encoder (K, V). The output feature sequence of the target encoder (K) is then attended by

[1]We did not compare with the supervised speaker representations because the recent work [19] showed that they are not suitable for delivering speaker information needed in VC.

http://dx.doi.org/10.21437/Interspeech.2021-1356

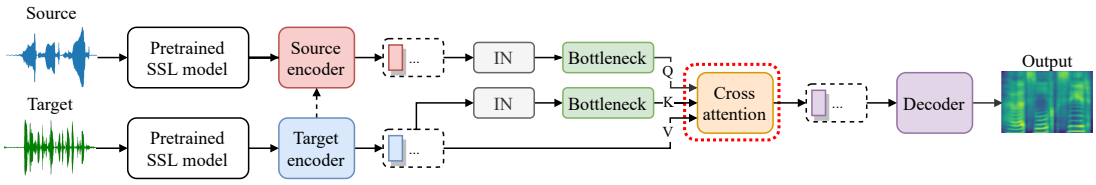

Figure 1: *The overall model architecture of S2VC. IN denotes the Instance Normalization.*

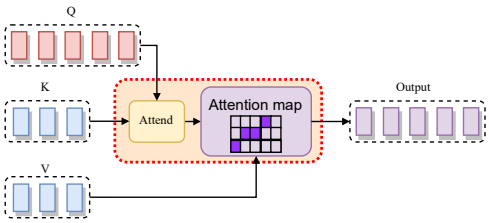

Figure 2: *Conceptual illustration of cross attention.*

the source encoder's output (Q). In this architecture, it is found that the cross attention module learns to align the source features to the target features with similar phonetic content, which is similar to the idea of exemplar-based VC models. The decoder then produces the converted Mel spectrogram from the attention-augmented features (V). At training time, the same utterance is used as the input of both the source and the target encoder and the decoder's reconstruction target. The encoders automatically learn to disentangle the content and the speaker information without any explicit constraint.

2.2. Modifications

We try to help the cross attention module to align the source and target features from the following two perspectives. Self-attention pooling guides the representation encoded by the source encoder to be close to that encoded by the target encoder. Attention information bottleneck removes redundant information in both representations encoded in Q and K to make the attention only consider the phonetic content information.

2.2.1. Self-attention pooling

Safar et al. [20] proposed a self-attention pooling layer and showed it is good at extracting the time-invariant features information. We utilize the self-attention pooling layer to extract the representation from the target encoder. The extracted representation is then applied to the source encoder (dotted line in Fig. 1) to guide the representation encoded by the source encoder closer to that by the target encoder.

2.2.2. Attention information bottleneck

AdaINVC [14] showed that Instance Normalization can remove speaker-dependent information from the features, and AutoVC [15] used a carefully designed hidden dimension of the encoder layers to extract speaker-independent content information. We combined them to the attention layer by applying Instance Normalization to both the Q and K, followed by a bottleneck layer to lower the speaker information encoded in them.

2.3. SSL representations

Three well-known SSL representations studied in this work are APC [3, 21], CPC [17, 4] and wav2vec 2.0 [18]. APC learns the representation in a way similar to a conventional RNN-based language model. It takes Mel spectrograms as input, and by predicting the future input conditioning on the past inputs, APC learns general speech representations in an auto-regressive way. On the other hand, CPC and wav2vec 2.0 directly utilize waveforms as input. CPC also learns the representation in an auto-regressive way but the prediction is done in the compact latent space instead of the input feature space, and the training is to optimize a probabilistic contrastive loss. Wav2vec 2.0 further improves on CPC by enlarging the model size and replacing auto-regressive prediction with masked language model-like prediction similar to BERT.

In the original Fragment VC [12], the source encoder takes wav2vec 2.0 representations as inputs to extract a feature sequence containing content information of the source utterance, while the target encoder takes Mel spectrogram of several utterances of the target speaker as inputs. Though only two representations are studied in FragmentVC, we consider this architecture and training scheme to be very flexible to incorporate any kinds of speech representations. This paper takes the representation of APC, CPC, or wav2vec 2.0 as the input features of source and target encoders and explores all the combinations.

3. Experimental setup

3.1. Training Setup

We trained all the models on CSTR VCTK Dataset [22] with 44 hours of audios spoken by 109 native speakers. All the audios were resampled from 48k Hz to 16k Hz before extracting the features. The optimizer we used is AdamW [23] with learning rate 5e-5, and $\beta_1 = 0.9$, $\beta_2 = 0.999$. The source encoder is composed of 4 linear layers with batch normalization. The target encoder consists of 3 one-dimensional convolution layers. The decoder is comprised of 3 conformer [24] layers followed by a linear projection to the Mel spectrogram's dimension. The dimension of queries and keys in the cross attention is 4 for the models with bottleneck, and their dimension is 512 for those without bottleneck. The L1 loss between the predicted and the ground-truth log Mel spectrogram is used. Universal neural vocoding [25] trained on a combination of LJ-speech [26], LibriTTS-train-clean-100, and CMU Arctic dataset [27] for 200k steps with batch size 32 is adopted as the vocoder of all the models.

3.2. Feature Extraction

We extract both PPG and SSL representations via S3PRL[2] [28] speech toolkit, which provides a user-friendly interface to extract various pretrained representations. The PPG is pretrained on the TIMIT dataset [29], achieving 21.7% frame-wise phone error rate (much more challenging measurement than phone error rate) on the test set. As for the self-supervised features such as APC and CPC, the official pretrained checkpoints are used.

[2]https://github.com/s3prl/s3prl

3.3. Testing scenarios

We evaluate the voice conversion models in the following two scenarios. The first one is the conversion between speakers in the training dataset VCTK (s2s). The other one is the conversion between speakers in the unseen dataset CMU (u2u). For each scenario, we randomly sampled 400 testing pairs. Each testing pair contains one utterance from the source speaker and five utterances from the target speaker.

3.4. Objective evaluation

We adopt two automatic assessment systems to evaluate the quality and the speaker similarity of converted utterances. MOSNet [30] is adopted to efficiently assess the quality of the synthetic utterances. Similar to the Mean Opinion Score (MOS) scored by human subjects, MOSNet takes an utterance as input and outputs a score ranging from 1.0 to 5.0, where the higher the score is, the better the quality is.

A publicly available pretrained speaker verification (SV) system[3] is adopted to assess the speaker similarity between a converted utterance and a target utterance, as done in previous work [31]. The SV system first extracts the utterance-level embeddings of the converted utterance and the target utterance, and the cosine similarity between the embeddings is computed as the similarity score. The SV system accepts a converted utterance if the similarity score is higher than a pre-defined threshold which is computed by finding the EER throughout the dataset. The percentage of converted utterances accepted by the SV system, which we call SV accuracy, is used as an evaluation metric.

3.5. Subjective Evaluation

To evaluate the perceptual quality of converted utterances, we conducted Mean Opinion Score (MOS) tests in quality and speaker similarity. For the quality MOS test, the subjects listen to an authentic vocoder-reconstructed utterance or a converted utterance. Then they score it from 1 to 5 in terms of quality (1 means bad, and 5 means perfect). For the similarity MOS test, the subjects listen to an authentic target utterance and a converted utterance. Then they score it from 1 to 5 in terms of speaker similarity (1 means very different and 5 means absolutely the same) [31]. For models considered, we evaluate the same 40 pairs of real and converted utterances, sampled from the 400 testing pairs, with each scored by at least 5 subjects. The scores are reported with the 95% confidence intervals for each model. Since the u2u scenario is more challenging than that of s2s, the subjective evaluation is conducted for u2u.

4. Results and analysis

4.1. Objective performance analysis

Five representations, including Mel spectrogram, PPG, APC, CPC, and wav2vec 2.0, are used as the source or target feature of the VC models. The results of MOSNet predictions are listed in Table 1a. In this table, we compare the average MOSNet predictions of the models with different representations as source features to see the ability of the representations in providing content information. The results show that both APC and CPC outperform PPG, suggesting they may be promising to provide content information for VC. The performances of Mel spectrogram and wav2vec 2.0 are not ideal in the u2u scenario, showing that they are relatively not robust to unseen data.

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

The results of SV accuracies are listed in Table 1b. Here we compare the average SV accuracies of the models with different representations as target features to examine their capability in terms of providing speaker information. The results show that Mel spectrogram is the best choice for extracting speaker-dependent information; however, some models with CPC (with underline in table) achieve comparable or better performance than those with Mel spectrogram, showing that CPC is also good at providing speaker-dependent information for VC.

As the performance in the u2u scenario is considered more critical than s2s, Fig. 3 plots the overall objective results for u2u. Each model is denoted as A+B, which means taking A as the source feature and B as the target feature. The points of the two figures are the same, but with different colors. In the left and right figures, the colors represent the same source feature or target features are used, respectively. The closer to the upper right means a better model, which achieves better performance in terms of MOSNet prediction and SV results.

Among all models, we can see that the CPC+CPC performs the best, so we select it for the subjective analysis to further explore its ability. Aside from the CPC+CPC, three models are selected as the baselines in the following subjective analysis. The first one is Mel+Mel, as the Mel spectrogram has long been used in speech synthesis tasks and showed great performance in VC. The second one is PPG+Mel, because the PPG is believed to be a better choice in providing content information needed for VC than Mel-spectrogram and performed well in previous PPG-based VC. The last one is wav2vec 2.0+Mel, which is adopted in FragmentVC and showed excellent performance in VC.

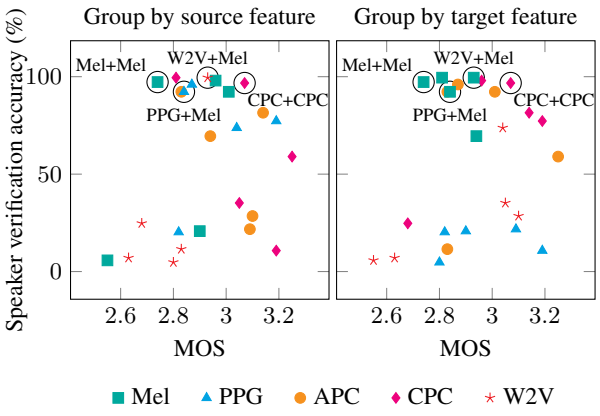

Figure 3: *The results of MOSNet prediction and Speaker Verification for u2u scenario.* W2V *denotes wav2vec 2.0.*

4.2. Subjective performance analysis

Subjective evaluation for quality and speaker similarity on the u2u scenario is conducted. The following models, CPC+CPC, Mel+Mel, PPG+Mel, and wav2vec 2.0+Mel are evaluated.

The results are listed in Table 2, suggesting that CPC+CPC outperforms other baseline models on both quality and speaker similarity, verified that CPC is suitable for providing both content and speaker information needed in VC. The converted audio samples are on the demo page [4] and the source code will be publicly available [5].

[4]https://howard1337.github.io/S2VC/
[5]https://github.com/howard1337/S2VC

Table 1: *The results of MOSNet prediction (a) and Speaker Verification (b) . A / B: s2s / u2u. W2V: wav2vec 2.0.*

(a)

Target	Source				
	Mel	PPG	APC	CPC	W2V
Mel	3.36 / 2.74	3.00 / 2.84	3.05 / 2.94	3.25 / 2.81	3.44 / 2.93
PPG	2.94 / 2.90	2.46 / 2.82	3.06 / 3.09	3.30 / 3.19	2.95 / 2.80
APC	3.07 / 3.01	3.06 / 2.87	2.97 / 2.83	3.41 / 3.25	3.03 / 2.83
CPC	3.29 / 2.96	3.16 / 3.19	2.83 / 3.14	3.36 / 3.07	3.08 / 2.68
W2V	2.95 / 2.55	2.99 / 3.04	3.12 / 3.10	3.30 / 3.05	2.93 / 2.63
Average	3.12 / 2.83	2.93 / 2.95	3.01 / 3.02	**3.32 / 3.07**	3.09 / 2.77

(b)

Source	Target				
	Mel	PPG	APC	CPC	W2V
Mel	93.0 / 97.3	12.3 / 20.8	69.0 / 92.3	98.8 / 98.0	14.3 / 5.8
PPG	78.8 / 95.3	14.5 / 20.3	81.3 / 96.0	78.8 / 88.3	69.0 / 73.8
APC	69.5 / 89.0	11.8 / 21.8	66.5 / 92.3	53.5 / 81.5	19.3 / 28.5
CPC	98.8 / 99.5	19.3 / 10.8	90.0 / 59.0	97.8 / 96.8	68.0 / 35.3
W2V	96.3 / 99.5	6.3 / 4.8	38.3 / 11.5	41.3 / 24.8	31.5 / 7.0
Average	**87.3 / 96.1**	12.8 / 15.7	69.0 / 70.2	74.0 / 77.9	40.4 / 32.1

Table 2: *The MOS on unseen-to-unseen conversion.*

MOS	Mel+Mel	PPG+Mel	W2V+Mel	CPC+CPC	Auth.
Sim.	2.97±0.19	3.05±0.20	3.16±0.20	3.33±0.21	–
Nat.	2.62±0.15	3.21±0.18	2.69±0.16	3.52±0.17	4.38±0.17

A+B: model with A as source feature and B as target feature.
W2V: wav2vec 2.0.
Auth.: vocoder-reconstructed authentic utterances.

4.3. Speaker information probing analysis

As the CPC+CPC model performs well in both objective and subjective evaluation, here we conduct the speaker information probing analysis on the query(Q), key(K), and value(V) for models taking CPC as source feature or as target features. The speaker classification (SC) task is adopted on the VCTK dataset as the probing task for speaker information. We randomly sampled 90% of the VCTK as the training set of SC and the rest 10% as the development set, and we adopt a linear layer after the extracted feature to classify the speakers. The training objective for the query feature is to predict the source speaker, while the objective for key and value features is to predict the target speaker. Both source and target encoder will be used in this analysis, and the source and target feature are ensured to be from different speakers to simulate the inference scenario.

The results are listed in Table 3. The SC accuracy of the query and key features are extremely low, showing that the Instance Normalization and the bottleneck layer can effectively remove the speaker-dependent information. As for the SC accuracy of the value feature, the models taking CPC as target feature perform better against other models taking PPG, APC, or wav2vec 2.0, showing that CPC can provide rich speaker information needed for VC.

Table 3: *Speaker information probing for model taking CPC an source or target feature.*

Source	Target	SC (Query)	SC (Key)	SC (Value)
		Dev(%)	Dev(%)	Dev(%)
PPG	**CPC**	2.83	3.19	91.87
APC		2.69	3.68	91.25
CPC		3.36	3.26	92.08
W2V		3.31	3.77	91.36
CPC	PPG	2.74	2.09	7.15
	APC	2.72	5.12	90.25
	CPC	2.36	3.26	92.08
	W2V	2.25	2.11	70.29

4.4. Ablation analysis

We conduct the ablation analysis on the CPC+CPC model. The SOTA approach FragmentVC [12] is considered as a baseline.

The ablation results are listed in Table 4. It suggests that CPC+CPC outperforms FragmentVC on all metrics considered. Rows (c) (d) (e) (f) (g) are respectively for removing the self-attention layer, removing the bottleneck layer, removing the instanceNorm layer, removing both bottleneck and instanceNorm layer, and removing cross attention, namely the decoder directly take the output (Q) of the source encoder. The results verified that all of them are essential for the framework.

Table 4: *Ablation study on the self-attention pooling, bottleneck layer, Instance Normalization and cross attention.*

Models	MOSNet		SV	
	S2S	U2U	S2S(%)	U2U(%)
(a) *FragmentVC [12]	3.14	3.00	89.00	91.75
(b) *Proposed	3.36	3.07	97.75	96.75
(c) *-SAP	3.21	2.78	97.00	95.00
(d) *-Bottleneck	2.90	2.49	88.00	93.25
(e) *-InstanceNorm	3.26	2.77	98.75	95.25
(f) *-Bottleneck, InstanceNorm	2.91	2.48	88.5	95.00
(g) *-Cross attention	3.55	3.22	31.25	25.75

FragmentVC here uses the officially released checkpoint.

5. Conclusion

We investigated several SSL representations to improve VC. We found that the model taking CPC as both source and target features outperform the baseline models on both subjective and objective evaluation, including a strong baseline model using PPG as source feature and Mel spectrogram as target feature. The results suggest that SSL representation CPC is suitable for providing both content and speaker information needed in VC. Furthermore, the ablation analysis showed that the proposed framework achieves comparable or even better performance than the SOTA approach FragmentVC [12] on objective evaluation. What will happen if we concatenate several different features like PPG and CPC as source feature and with other combinations of representations as target feature are yet to be investigated. We believe that different representations complement each other and provide richer information for both content and speaker information to gain further improvement.

6. Acknowledgements

We acknowledge the support of AWS Machine Learning Research Awards program.

7. References

[1] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. [Online]. Available: https://openreview.net/forum?id=S1v4N2l0-

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. [Online]. Available: https://www.aclweb.org/anthology/N19-1423

[3] Y. A. Chung and J. Glass, "Generative pre-training for speech with autoregressive predictive coding," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 3497–3501.

[4] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018.

[5] A. T. Liu, S. w. Yang, P. H. Chi, P. c. Hsu, and H. y. Lee, "Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6419–6423.

[6] 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)*, 2016, pp. 1–6.

[7] S. won Park, D. young Kim, and M. chul Joe, "Cotatron: Transcription-Guided Speech Encoder for Any-to-Many Voice Conversion Without Parallel Data," in *Proc. Interspeech 2020*, 2020, pp. 4696–4700. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2020-1542

[8] Z. Yi, W.-C. Huang, X. Tian, J. Yamagishi, R. K. Das, T. Kinnunen, Z.-H. Ling, and T. Toda, "Voice Conversion Challenge 2020 — Intra-lingual semi-parallel and cross-lingual voice conversion —," in *Proc. Joint Workshop for the Blizzard Challenge and Voice Conversion Challenge 2020*, 2020, pp. 80–98. [Online]. Available: http://dx.doi.org/10.21437/VCC_BC.2020-14

[9] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4879–4883.

[10] 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.

[11] W.-C. Huang, Y.-C. Wu, T. Hayashi, and T. Toda, "Any-to-one sequence-to-sequence voice conversion using self-supervised discrete speech representations," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021.

[12] Y. Y. Lin, C.-M. Chien, J.-H. Lin, H. yi Lee, and L. shan Lee, "Fragmentvc: Any-to-any voice conversion by end-to-end extracting and fusing fine-grained voice fragments with attention," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021.

[13] S. Liu, J. Zhong, L. Sun, X. Wu, X. Liu, and H. Meng, "Voice conversion across arbitrary speakers based on a single target-speaker utterance," in *Proc. Interspeech 2018*, 2018, pp. 496–500. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1504

[14] J. chieh Chou and H.-Y. Lee, "One-Shot Voice Conversion by Separating Speaker and Content Representations with Instance Normalization," in *Proc. Interspeech 2019*, 2019, pp. 664–668. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-2663

[15] K. Qian, Y. Zhang, S. Chang, X. Yang, and M. Hasegawa-Johnson, "Autovc: Zero-shot voice style transfer with only autoencoder loss," in *Proceedings of the 36th International Conference on Machine Learning*, 2019, pp. 5210–5219. [Online]. Available: http://proceedings.mlr.press/v97/qian19c.html

[16] D.-Y. Wu, Y.-H. Chen, and H. yi Lee, "VQVC+: One-Shot Voice Conversion by Vector Quantization and U-Net Architecture," in *Proc. Interspeech 2020*, 2020, pp. 4691–4695.

[17] M. Rivière, A. Joulin, P. E. Mazaré, and E. Dupoux, "Unsupervised pretraining transfers well across languages," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7414–7418.

[18] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020.

[19] T. h. Huang, J. h. Lin, and H. y. Lee, "How far are we from robust voice conversion: A survey," in *2021 IEEE Spoken Language Technology Workshop (SLT)*, 2021, pp. 514–521.

[20] P. Safari, M. India, and J. Hernando, "Self-Attention Encoding and Pooling for Speaker Recognition," in *Proc. Interspeech 2020*, 2020, pp. 941–945. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2020-1446

[21] Y.-A. Chung, W.-N. Hsu, H. Tang, and J. Glass, "An Unsupervised Autoregressive Model for Speech Representation Learning," in *Proc. Interspeech 2019*, 2019, pp. 146–150. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-1473

[22] C. Veaux, J. Yamagishi, K. MacDonald *et al.*, "Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit," 2017.

[23] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=Bkg6RiCqY7

[24] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented Transformer for Speech Recognition," in *Proc. Interspeech 2020*, 2020, pp. 5036–5040. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2020-3015

[25] J. Lorenzo-Trueba, T. Drugman, J. Latorre, T. Merritt, B. Putrycz, R. Barra-Chicote, A. Moinet, and V. Aggarwal, "Towards Achieving Robust Universal Neural Vocoding," in *Proc. Interspeech 2019*, 2019, pp. 181–185.

[26] K. Ito and L. Johnson, "The lj speech dataset," https://keithito.com/LJ-Speech-Dataset/, 2017.

[27] J. Kominek and A. W. Black, "The cmu arctic speech databases," in *Fifth ISCA workshop on speech synthesis*, 2004.

[28] A. T. Liu and Y. Shu-wen, "S3prl: The self-supervised speech pre-training and representation learning toolkit," 2020. [Online]. Available: https://github.com/s3prl/s3prl

[29] J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallett, N. Dahlgren, and V. Zue, "TIMIT Acoustic-Phonetic Continuous Speech Corpus," 1993. [Online]. Available: https://hdl.handle.net/11272.1/AB2/SWVENO

[30] 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," in *Proc. Interspeech 2019*, 2019, pp. 1541–1545. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-2003

[31] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, z. Chen, P. Nguyen, R. Pang, I. Lopez Moreno, and Y. Wu, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *Advances in Neural Information Processing Systems 31*. Curran Associates, Inc., 2018, pp. 4480–4490. [Online]. Available: http://papers.nips.cc/paper/7700-transfer-learning-from-speaker-verification-to-multispeaker-text-to-speech-synthesis.pdf