Prosody-Adaptable Audio Codecs for Zero-Shot Voice Conversion via In-Context Learning

Junchuan Zhao[1*], *Xintong Wang*[1*], *Ye Wang*[1]

[1]School of Computing, National University of Singapore, Singapore

junchuan@u.nus.edu, xintong.wang@u.nus.edu, dcswangy@nus.edu.sg

Abstract

Recent advances in discrete audio codecs have significantly improved speech representation modeling, while codec language models have enabled in-context learning for zero-shot speech synthesis. Inspired by this, we propose a voice conversion (VC) model within the VALLE-X framework, leveraging its strong in-context learning capabilities for speaker adaptation. To enhance prosody control, we introduce a prosody-aware audio codec encoder (PACE) module, which isolates and refines prosody from other sources, improving expressiveness and control. By integrating PACE into our VC model, we achieve greater flexibility in prosody manipulation while preserving speaker timbre. Experimental evaluation results demonstrate that our approach outperforms baseline VC systems in prosody preservation, timbre consistency, and overall naturalness, surpassing baseline VC systems.

Index Terms: voice conversion; audio codec; prosody control

1. Introduction

Voice conversion (VC) is an advanced speech processing technique that facilitates the transformation of one speaker's voice into another's while preserving the underlying content [1, 2]. By effectively separating linguistic content from speaker-specific features such as timbre and prosody, VC enables the synthesis of speech that preserves the original message while adopting the vocal characteristics of a target speaker. This technology has gained significant attention for its diverse applications, enhancing communication and user experience. VC is vital to preserve privacy through voice anonymization and empowers individuals with speech impairments to express their desired vocal identity. Its ability to manipulate voice attributes makes VC a transformative tool in speech synthesis and human-computer interaction, driving innovation and improving accessibility in various domains [3].

Recent advances in deep learning have transformed VC by improving the quality of synthesized voices, leading to more natural intonation, clearer articulation, and greater emotional expression for a more authentic auditory experience [4, 5, 6, 7, 8, 9]. However, two major challenges still present opportunities for further advancement. The first is to achieve robust disentanglement between speech content and speech features, as well as between different features such as prosody and timbre, to enable more precise control over voice characteristics. The second challenge lies in improving zero-shot voice conversion, where the system must generate high-quality voice transformations for unseen speakers without requiring extensive training data.

Most VC systems focus primarily on disentangling content from speaker-specific features [10, 11, 12], but disentangling within the speaker features themselves, such as separating prosody, timbre, and pitch, is equally critical. Achieving this finer level of separation is essential for enabling more expressive voice conversion and providing greater control over voice characteristics, allowing for nuanced transformations that capture the full range of human speech dynamics. There are several studies have concentrated on disentangling prosody and timbre for voice conversion [13, 14, 15]. [14] introduces an innovative method that employs a unit encoder, speaker verification, and a prosody encoder, enhanced by an adversarial content predictor. This approach effectively minimizes information overlap between prosody and content embedding, facilitating more distinct and controlled representations. [15] introduces a self-supervised method that learns disentangled pitch and volume representations from augmented speech, effectively capturing prosody styles and enhancing zero-shot voice conversion while mitigating prosody leakage.

Zero-shot voice conversion remains a key challenge due to the need for systems to adapt to unseen speakers without prior training. Previous studies have employed speaker embeddings to capture timbre information from reference speakers, enabling systems to generalize to new voices without fine-tuning, thereby improving the adaptability of voice conversion technologies across diverse speaker identities [16, 17]. However, a major limitation of this approach in both zero-shot voice conversion and text-to-speech (TTS) is the dependency on a robust, well-trained speaker encoder, which requires access to a large and diverse dataset to perform effectively. Recently, researchers have explored the potential of in-context learning (ICL) to overcome this challenge in TTS. By employing a target speech prompting strategy, ICL enables systems to generate speech for previously unseen speakers without the need for a pretrained speaker encoder [16, 17]. This innovative approach significantly enhances zero-shot performance by bypassing the requirement for explicit speaker embeddings, making TTS systems more flexible and scalable.

In this paper, we present a novel zero-shot VC system that combines disentangled prosody control with the ICL capabilities of advanced pretrained models. Our proposed system aims to achieve high speaker similarity and preservation of prosody in voice conversion. Specifically, (1) we achieve explicit prosody disentanglement from other speech attributes (content and timbre) by proposing a Prosody-Aware Codec Encoder (PACE), enabling finer control over expressive variations. (2) We leverage the pretrained VALL-E X model [18], a well-performed TTS system with emergent ICL capabilities, as the backbone of our VC system. This allows for high-quality speech generation while preserving key speaker attributes, even

[*]These authors contributed equally to this research.

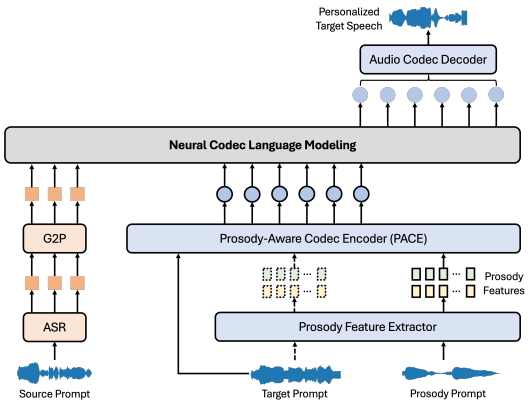

Figure 1: *Overall architecture of the proposed voice conversion system. Dotted lines denote components used only during training. Prosody features are derived from the prosody prompt during inference.*

for unseen speakers. (3) To align PACE-generated audio codes with those of VALL-E X, we train the PACE module using VALL-E X audio codes as targets. Our results show that our VC system outperforms baselines in speech quality, timbre similarity, and prosody controllability, enabling a high-quality zero-shot VC system that preserves both speaker identity and prosodic consistency.

2. Methodology

2.1. Overall Architecture

The proposed voice conversion (VC) system, depicted in Figure 1, is built upon VALL-E X [18], a state-of-the-art (SOTA) end-to-end text-to-speech (TTS) model. VALL-E X has demonstrated strong generalization across diverse languages and speech tasks, particularly in zero-shot cross-lingual TTS and speech-to-speech translation (S2ST). The original VALL-E X framework processes both a text prompt and a speech prompt as inputs. The text prompt is transcribed into a phoneme sequence using a grapheme-to-phoneme (G2P) module, while the audio codec encoder [19] maps the speech prompt to an embedding representation. This embedding undergoes residual vector quantization (RVQ) to derive the corresponding audio codes. The neural codec language model then autoregressively predicts audio codes conditioned on the phoneme sequence and speech codes, and the audio codec decoder subsequently synthesizes the output speech waveform.

To exploit the emergent in-context learning (ICL) capabilities of VALL-E X in timbre modeling for VC, we adapt the model for this task. In the VC setting, the model takes as input a source speech prompt $\mathbf{w}^s \in \mathbb{R}^{L_s \times 1}$ and a target prompt $\mathbf{w}^t \in \mathbb{R}^{L_t \times 1}$. The objective is to generate the converted speech $\mathbf{o} \in \mathbb{R}^{L_o \times 1}$, where L_s, L_t, and L_o denote the lengths of the source prompt, target prompt, and output, respectively, and the second dimension 1 indicates monaural audio. The generated speech retains the linguistic content of the source speech prompt while adopting the style of the target prompt.

For the source speech prompt, we use Whisper-Medium [20] for automatic speech recognition (ASR) to transcribe the speech into text, which is then processed by a grapheme-to-phoneme (G2P) module to obtain the phoneme sequence $\mathbf{p} \in \{0, \ldots, N-1\}^S$, where S is the sequence length and N is the phoneme vocabulary size.

According to the target speech prompt, a straightforward approach is to directly input it into the audio codec encoder to obtain speech codes. However, this approach lacks prosody control. To address this, we extract prosodic features from the target prompt, including fundamental frequency (f_0) and unvoiced/voiced (uv) indicators, following [21, 22, 23]. We further introduce the Pitch-Aware Codec Encoder (PACE) module, which derives speech codes conditioned on these prosody features. The PACE module is trained using the extracted prosody features alongside the target speech prompt.

2.2. Prosody-Aware Codec Encoder (PACE)

The Prosody-Aware Codec Encoder (PACE) module, as shown in Figure 2, is designed to extract the audio codec representation from the target speech prompt while conditioning on prosody features. PACE is built upon the audio codec encoder from SoundStream [19] but is structurally modified by splitting the encoder into two stages. The original codec encoder consists of four convolutional blocks, downsampling speech length by factors of $[2, 4, 5, 8]$; we retain the first three layers to extract the speech embedding $\mathbf{e}^f \in \mathbb{R}^{\frac{L}{40} \times D}$, where D denotes the embedding dimension, and $\frac{L}{40}$ represents the sequence length after downsampling.

To disentangle the prosody information from the $\mathbf{e}^f$, we minimize the mutual information (MI) estimation between $\mathbf{e}^f$ and the prosody embeddings $\mathbf{e}^{f_0}$, $\mathbf{e}^{uv}$ through contrastive log-ratio upper bound (CLUB), following [24, 25, 26]. We first extract the prosody features (f_0, uv) via a Prosody Feature Extractor. Specifically, we employ the harvest function from pyworld[1] to compute f_0 and uv with a frame shift of $\frac{40}{24000} \times 1000 = \frac{5}{3}$ ms, ensuring length alignment with $\mathbf{e}^f$. The f_0 sequence is then normalized to $[0, 1]$, quantized into 256 discrete values, and represented as $f_0 \in \{0, 1, \cdots, 255\}^{\frac{L}{40}}$, while uv is as $uv \in \{0, 1\}^{\frac{L}{40}}$. The MI minization estimation for both f_0 and uv is shown in Equation 1 and 2.

$$\begin{aligned}\mathcal{L}_{MI}^{f_0} &= \mathbb{E}_{p(\mathbf{e}^{f_0}, \mathbf{e}^f)}\left[\log q\left(\mathbf{e}^{f_0} \mid \mathbf{e}^f\right)\right] \\ &\quad - \mathbb{E}_{p(\mathbf{e}^{f_0})p(\mathbf{e}^f)}\left[\log q\left(\mathbf{e}^{f_0} \mid \mathbf{e}^f\right)\right] \\ &= \frac{1}{N}\sum_{i=1}^{N}\left[\log q_\theta\left(\mathbf{e}_i^{f_0} \mid \mathbf{e}_i^f\right) - \frac{1}{M}\sum_{j=1}^{M}\log q_\theta\left(\mathbf{e}_j^{f_0} \mid \mathbf{e}_i^f\right)\right], \end{aligned} \tag{1}$$

$$\begin{aligned}&\mathcal{L}_{MI}^{uv} = \\ &\frac{1}{N}\sum_{i=1}^{N}\left[\log q_\theta\left(\mathbf{e}_i^{uv} \mid \mathbf{e}_i^f\right) - \frac{1}{M}\sum_{j=1}^{M}\log q_\theta\left(\mathbf{e}_j^{uv} \mid \mathbf{e}_i^f\right)\right]. \end{aligned} \tag{2}$$

After obtaining the prosody-invariant audio embedding $\mathbf{e}^f$, we incorporate the prosody information back by summing it with the prosody embeddings $\mathbf{e}^{f0}$ and $\mathbf{e}^{uv}$. These embeddings are derived by passing f_0 and uv through an embedding layer.

To ensure the generated audio codes align with those used in the neural codec language model, we employ the trained codec encoder from VALLE-X to first generate the target audio embedding $\mathbf{e}^c \in \mathbb{R}^{\frac{L}{320} \times D'}$, which serves as the input to the residual vector quantizer (RVQ), where D' denotes the embedding dimension, and $\frac{L}{320}$ results from downsampling $\frac{L}{40}$ by

[1] https://github.com/JeremyCCHsu/Python-Wrapper-for-World-Vocoder

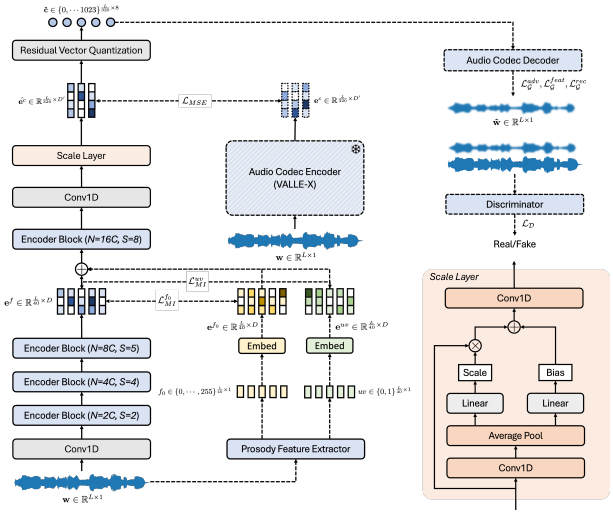

Figure 2: *Architecture of proposed Prosody-Aware Codec Encoder (PACE) module. Dotted lines denote components used only during training. Prosody features and embeddings are derived from the prosody prompt during inference.*

a factor of 8. We propose a scale layer to align the values of the predicted $\hat{\mathbf{e}^c}$ and target $\mathbf{e}^c$ embeddings within a similar range. This scale layer consists of a Conv1D layer and an Average Pooling layer, which predict a scaling factor K and a bias term B. These values are then applied to scale the audio codec embedding, which is subsequently passed through another Conv1D layer, as illustrated in Equation 3 to obtain the scaled audio codec embedding $\hat{\mathbf{e}^c}$.

$$\hat{\mathbf{e}^c} = \text{Conv1D}(K \times \tilde{\mathbf{e}^c} + B), \tag{3}$$

where $\tilde{\mathbf{e}^c}$ denotes the audio codec embedding before scaling. To ensure accurate reconstruction, we minimize the mean squared error (MSE) loss between the predicted and target audio codec embeddings as defined Equation 4,

$$\mathcal{L}^e_{recon} = \mathcal{L}_{MSE}(\hat{\mathbf{e}^c}, \mathbf{e}^c). \tag{4}$$

The generated audio codes $\hat{\mathbf{c}} \in \{0, 1, \cdots, 1023\}^{\frac{L}{320} \times 8}$, where 8 represents the number of codebooks used, are then obtained by passing the predicted audio embedding $\hat{\mathbf{e}^c} \in \mathbb{R}^{\frac{L}{320} \times D'}$ through the RVQ, conditioned on the prosody features.

Beyond the mutual information loss, we adopt the the adversarial loss $\mathcal{L}_{adv}$, the feature loss $\mathcal{L}_{feat}$, and the multi-scale spectral reconstruction loss for generator, $\mathcal{L}_{rec}$ following [19]. These losses jointly guide the training of the PACE module, facilitating high-quality speech generation. Additionally, we follow [19] in employing a discriminator $\mathcal{D}$ with the corresponding loss $\mathcal{L}^{\mathcal{D}}$ to enhance perceptual quality. The overall objective for generator loss of the PACE module is formulated in Equation 5,

$$\begin{aligned}\mathcal{L}_{\mathcal{G}} = \lambda_{MI}\mathcal{L}_{MI} + \lambda^e_{recon}\mathcal{L}^e_{recon} + \lambda_{adv}\mathcal{L}_{adv} \\ + \lambda_{feat}\mathcal{L}_{feat} + \lambda_{rec}\mathcal{L}_{rec},\end{aligned} \tag{5}$$

where $\mathcal{L}_{MI} = \mathcal{L}^{f_0}_{MI} + \mathcal{L}^{uv}_{MI}$.

2.3. Training and Inference Scheme

The model training follows a three-stage process. In the first stage, the PACE module is trained without $f0$ and uv as input by minimizing the loss of reconstruction of the audio codec embedding $\mathcal{L}^e_{rec}$. In the second stage, the prosody information is disentangled using the codec encoder parameters learned in stage one, optimizing mutual information loss $\mathcal{L}_{MI}$ along with $\mathcal{L}^e_{rec}$. Finally, in the third stage, the PACE encoder, audio codec decoder are jointly trained by optimizing the total loss.

During the inference phrase, we first extract the phoneme sequence $\mathbf{p}$ from the source prompt $\mathbf{w^s}$ with speaker A. Next, we obtain the prosody features (f_0, uv) from the prosody prompt $\mathbf{w^P}$ (which can be any speech prompt) using the Prosody Feature Extractor, with speaker X (can be A and B). These prosody features, along with the target prompt $\mathbf{w^t}$ (from speaker B), are then input into the PACE module to generate the prosody-adaptable speech codes $\hat{\mathbf{c}}$. Subsequently, the neural codec language model is utilized to generate the target speech codes $\mathbf{c^o}$, which are passed through the audio codec decoder to produce the target speech $\mathbf{o}$. This generated speech $\mathbf{o}$ retains the content of $\mathbf{w^s}$, the timbre of $\mathbf{w^t}$, and the prosody of $\mathbf{w^P}$.

3. Experiments

3.1. Training Setup

We train our model on a 54-hour LibriTTS-clean-100 dataset [27] and evaluate it on the test-clean set. The training set consists of 33,236 speech samples from 247 speakers, all resampled to 24 kHz. For zero-shot evaluation, we select 10 male and 10 female speakers from LibriTTS. During training, we randomly extract 2-second segments from the speech clips, with zero-padding applied to clips shorter than 2 seconds.

The encoder and decoder of PACE module follows the architecture in [19]. The f_0 and uv embedding layers have a dimension of 256, with vocabulary sizes of 256 and 2. The scale layer includes a Conv1D layer $(128, 64)$ with a kernel size of 3 for scale and bias extraction, followed by a linear layer $(64, 1)$, and a final Conv1D layer $(128, 128)$ with a kernel size of 3. Our experiments are implemented in PyTorch and PyTorch Lightning, our model was trained on NVIDIA RTX A5000 GPUs for 360k, 60k, and 180k steps across the three training stages.

3.2. Evaluation Methods

For evaluation, we use the following metrics. The ASV-Score [18] measures speaker similarity by calculating the cosine distance between speaker embeddings from a pretrained WavLM model. The ASR-WER [20] evaluates intelligibility by comparing synthesized speech to ground truth transcriptions using the Whisper model. We assess naturalness with the NISQA-TTS model[2] [28], providing a score from 0 to 5 for fluency, clarity, and expressiveness. The MOSNET model[3] [29] offers an objective MOS score, reflecting overall speech quality. For prosody matching, we compute the F0 distance [30], measured as the normalized distance between the F0 contours of converted and reference speech. Subjective evaluation involved mean opinion scores (MOS) and speaker similarity (SMOS), with 22 participants rating audio samples from various systems on a scale of 1 to 5, where 5 represented the highest quality.

3.3. Effectiveness of Pitch-Aware Codec Encoder (PACE) module

We compare PACE module with the pretrained 24 kHz EnCodec[4], the audio codec encoder originally used in VALL-E X

[2]https://github.com/gabrielmittag/NISQA
[3]https://github.com/aliutkus/speechmetrics
[4]https://huggingface.co/facebook/encodec_24khz

Table 1: *Evaluation for PACE v.s. Baseline encodec encoder.*

Model	ASV (↑)	ASR-WER (↓)	NISQA (↑)	MOSNET (↑)
Baseline Codec Encoder	0.6813	0.1239	4.1662	3.8755
PACE	0.6620	0.1104	3.9805	3.6592
Ground Truth	-	-	4.6382	4.0233

Table 2: *Objective and Subjective Evaluation of Proposed and Baseline VC Systems.*

Model	ASV (↑)	ASR-WER (↓)	NISQA (↑)	MOSNET (↑)	MOS (↑)	SMOS (↑)
VALLE-X	0.8429	0.1151	4.3034	3.6059	4.1880	3.8251
TriAAN-VC	0.7199	0.1298	4.2527	3.5451	4.0667	3.7870
Proso-VC	0.7025	0.1632	3.2520	2.9105	3.9520	3.2542
Ours	0.9078	0.1010	4.3320	3.5746	4.3627	3.9386
-w/o $\mathcal{L}_{MI}$	0.8751	0.1314	4.2892	3.4178	3.7696	3.4074
-w/o scale layer	0.8771	0.1574	3.9424	2.7236	3.2876	2.8333
-w/o $\mathcal{L}^e_{rec}$	0.5882	0.2286	3.0974	2.0822	2.4831	2.3550
Ground Truth	-	-	4.5890	3.9328	4.3940	4.1585

(Baseline Codec Encoder), across objective evaluation metrics. As shown in Table 1, while there remains a slight gap in performance between the original EnCodec and our proposed PACE module, this difference does not significantly impact overall performance. This suggests that our PACE module effectively retains much of the original EnCodec's capabilities. Notably, the ASR-WER demonstrates a substantial improvement, reflecting enhanced speech intelligibility.

3.4. Voice Conversion: Speaker Timbre Control and Overall Quality

We evaluated the overall quality of voice conversion by comparing our model with baseline models: VALLE-X, TriAAN-VC, and ProsoVC, as shown in Table 2. TriAAN-VC [31] enables nonparallel conversion from any to any through adaptive attention normalization, while ProsoVC [14] controls prosody using hybrid bottleneck features. Our model outperforms these baselines in most metrics, achieving the highest ASV score, a lower ASR-WER, and the best NISQA score, indicating superior preservation, intelligibility, and naturalness of the speaker identity. Although the MOSNET score is slightly lower than the VALLE-X, it is still higher than other baselines, underscoring our model's overall effectiveness in voice conversion. Ablation studies demonstrate the effectiveness of our method, significantly outperforming the model without the scale layer and reconstruction loss training, while achieving a slight improvement over the variant without mutual information loss training. Subjective evaluation results indicate that our model achieved strong performance in both naturalness and speaker similarity, particularly outperforming ablated variants, demonstrating the effectiveness of the proposed modules.

3.5. Voice Conversion: Capability in Prosody Alignment

We evaluate prosody matching in two scenarios: prosody from the source speech and from a prompt. Results in Table 3 are compared with VALLE-X, TriAAN-VC, and ProsoVC using normalized F0 distance. VALLE-X is excluded from the "source" scenario, and TriAAN-VC/ProsoVC from the "prompt" scenario, as they lack these functions.

Our model outperforms the baselines in both scenarios. In the "prosody from source" case, it leads, while VALLE-X, limited to prompt-based prosody conversion, cannot perform in this scenario. In the "prosody from prompt" case, VALLE-X shows reasonable performance due to the ICL capability but still lags behind our model, emphasizing the effectiveness of our disentanglement approach for prosody matching. We visualize the normalized F0 curves of the synthesized output and the reference prosody prompt. As shown in Figure 3, our method's F0 curve closely aligns with the reference, demonstrating its effective prosody control. In ablation studies, our model outperforms the ablated variants in both scenarios, with a more pronounced advantage in the "Prosody from Source" setting. While the model trained without $\mathcal{L}_{MI}$ achieves comparable performance in overall quality and timbre similarity, the results highlight the importance of mutual information loss in prosody control.

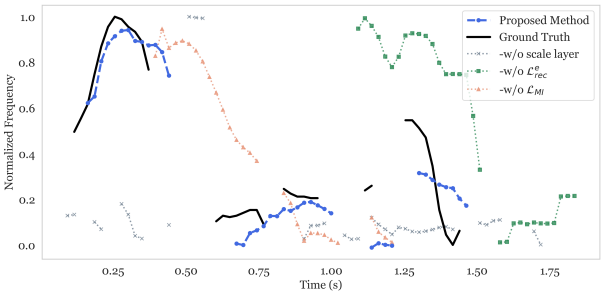

Figure 3: *F0 contour comparison for our proposed method and reference prosody prompt.*

Table 3: *Prosody Matching in Voice Conversion (F0-scaled Distance).*

Models	Prosody from Source	Prosody from Target
VALL-E X	-	3.1029
TriAAN-VC	3.4019	-
ProsoVC	3.3256	-
Ours	2.8239	2.6988
-w/o $\mathcal{L}_{MI}$	3.2751	3.0179
-w/o scale layer	3.9178	3.6237
-w/o $\mathcal{L}^e_{rec}$	4.7892	4.2649

4. Conclusion

We propose a voice conversion (VC) model that explicitly disentangles prosody from speaker timbre, enabling precise prosody control while preserving speaker identity. To achieve this, we introduce the prosody-aware audio codec encoder (PACE), which conditions audio codes on specific prosodic features, allowing fine-grained manipulation of prosody. By isolating prosody from other acoustic attributes, our approach enhances prosody control, ensuring greater flexibility in voice conversion tasks. Comprehensive evaluation results demonstrate that our model outperforms baseline systems in prosody alignment, timbre consistency, and overall speech quality. In particular, our method achieves more natural and expressive voice conversion that closely matches the target speaker's style. These results underscore the effectiveness of our disentanglement strategy and its broader applicability in controllable speech synthesis and expressive voice conversion.

5. References

[1] S. Liu, Y. Cao, S. Kang, N. Hu, X. Liu, D. Su, D. Yu, and H. Meng, "Transferring source style in non-parallel voice conversion," in *21st Annual Conference of the International Speech Communication Association, Interspeech 2020*, 2020, pp. 4721–4725.

[2] B. Sisman, J. Yamagishi, S. King, and H. Li, "An overview of voice conversion and its challenges: From statistical modeling to deep learning," *IEEE ACM Trans. Audio Speech Lang. Process.*, vol. 29, pp. 132–157, 2021.

[3] T. Walczyna and Z. Piotrowski, "Overview of voice conversion methods based on deep learning," *Applied Sciences*, vol. 13, no. 5, 2023.

[4] Y. A. Li, A. Zare, and N. Mesgarani, "Starganv2-vc: A diverse, unsupervised, non-parallel framework for natural-sounding voice conversion," in *22nd Annual Conference of the International Speech Communication Association, Interspeech 2021*, 2021, pp. 1349–1353.

[5] E. Casanova, J. Weber, C. D. Shulby, A. C. Júnior, E. Gölge, and M. A. Ponti, "Yourtts: Towards zero-shot multi-speaker TTS and zero-shot voice conversion for everyone," in *International Conference on Machine Learning, ICML 2022*, vol. 162, 2022, pp. 2709–2720.

[6] V. Popov, I. Vovk, V. Gogoryan, T. Sadekova, M. S. Kudinov, and J. Wei, "Diffusion-based voice conversion with fast maximum likelihood sampling scheme," in *The Tenth International Conference on Learning Representations, ICLR 2022*. OpenReview.net, 2022.

[7] H. Tang, X. Zhang, J. Wang, N. Cheng, and J. Xiao, "Avqvc: One-shot voice conversion by vector quantization with applying contrastive learning," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022*, 2022, pp. 4613–4617.

[8] S. Lee, B. Ko, K. Lee, I. Yoo, and D. Yook, "Many-to-many voice conversion using conditional cycle-consistent adversarial networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020*, 2020, pp. 6279–6283.

[9] J. Lian, C. Zhang, and D. Yu, "Robust disentangled variational speech representation learning for zero-shot voice conversion," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022*, 2022, pp. 6572–6576.

[10] A. T. Liu, P. Hsu, and H. Lee, "Unsupervised end-to-end learning of discrete linguistic units for voice conversion," in *20th Annual Conference of the International Speech Communication Association, Interspeech 2019*, 2019, pp. 1108–1112.

[11] M. Luong and V. Tran, "Many-to-many voice conversion based feature disentanglement using variational autoencoder," in *22nd Annual Conference of the International Speech Communication Association, Interspeech 2021*, 2021, pp. 851–855.

[12] J. Chou and H. Lee, "One-shot voice conversion by separating speaker and content representations with instance normalization," in *20th Annual Conference of the International Speech Communication Association, Interspeech 2019*, 2019, pp. 664–668.

[13] Z. Lian, R. Zhong, Z. Wen, B. Liu, and J. Tao, "Towards fine-grained prosody control for voice conversion," in *12th International Symposium on Chinese Spoken Language Processing, ISCSLP 2021*, 2021, pp. 1–5.

[14] X. Zhao, F. Liu, C. Song, Z. Wu, S. Kang, D. Tuo, and H. Meng, "Disentangling content and fine-grained prosody information via hybrid ASR bottleneck features for voice conversion," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022*, 2022, pp. 7022–7026.

[15] S. Wang and D. Borth, "Zero-shot voice conversion via self-supervised prosody representation learning," in *International Joint Conference on Neural Networks, IJCNN 2022*, 2022, pp. 1–8.

[16] 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, ICML 2019*, vol. 97, 2019, pp. 5210–5219.

[17] R. Xiao, H. Zhang, and Y. Lin, "Dgc-vector: A new speaker embedding for zero-shot voice conversion," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022*, 2022, pp. 6547–6551.

[18] Z. Zhang, L. Zhou, C. Wang, S. Chen, Y. Wu, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li *et al.*, "Speak foreign languages with your own voice: Cross-lingual neural codec language modeling," *arXiv preprint arXiv:2303.03926*, 2023.

[19] N. Zeghidour, A. Luebs, A. Omran, J. Skoglund, and M. Tagliasacchi, "Soundstream: An end-to-end neural audio codec," *IEEE ACM Trans. Audio Speech Lang. Process.*, vol. 30, pp. 495–507, 2022.

[20] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *International Conference on Machine Learning, ICML 2023*, vol. 202, 2023, pp. 28 492–28 518.

[21] K. Byun, S. Moon, and E. Visser, "Highly controllable diffusion-based any-to-any voice conversion model with frame-level prosody feature," *arXiv preprint arXiv:2309.03364*, 2023.

[22] G. Pamisetty and K. S. R. Murty, "Prosody-tts: An end-to-end speech synthesis system with prosody control," *Circuits Syst. Signal Process.*, vol. 42, no. 1, pp. 361–384, 2023.

[23] X. An, F. K. Soong, S. Yang, and L. Xie, "Effective and direct control of neural TTS prosody by removing interactions between different attributes," *Neural Networks*, vol. 143, pp. 250–260, 2021.

[24] X. Chen, X. Xu, J. Chen, Z. Zhang, T. Takiguchi, and E. R. Hancock, "Speaker-independent emotional voice conversion via disentangled representations," *IEEE Trans. Multim.*, vol. 25, pp. 7480–7493, 2023.

[25] L. Huang, T. Yuan, Y. Liang, Z. Chen, C. Wen, Y. Xie, J. Zhang, and D. Ke, "LIMI-VC: A light weight voice conversion model with mutual information disentanglement," in *IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2023*, 2023, pp. 1–5.

[26] S. Yang, M. Tantrawenith, H. Zhuang, Z. Wu, A. Sun, J. Wang, N. Cheng, H. Tang, X. Zhao, J. Wang, and H. Meng, "Speech representation disentanglement with adversarial mutual information learning for one-shot voice conversion," in *23rd Annual Conference of the International Speech Communication Association, Interspeech 2022*, 2022, pp. 2553–2557.

[27] H. Zen, V. Dang, R. Clark, Y. Zhang, R. J. Weiss, Y. Jia, Z. Chen, and Y. Wu, "Libritts: A corpus derived from librispeech for text-to-speech," in *20th Annual Conference of the International Speech Communication Association, Interspeech 2019*, 2019, pp. 1526–1530.

[28] G. Mittag, B. Naderi, A. Chehadi, and S. Möller, "NISQA: A deep cnn-self-attention model for multidimensional speech quality prediction with crowdsourced datasets," in *22nd Annual Conference of the International Speech Communication Association, Interspeech 2021*, 2021, pp. 2127–2131.

[29] C. Lo, S. Fu, W. Huang, X. Wang, J. Yamagishi, Y. Tsao, and H. Wang, "Mosnet: Deep learning-based objective assessment for voice conversion," in *20th Annual Conference of the International Speech Communication Association, Interspeech 2019*, 2019, pp. 1541–1545.

[30] P. Lu, J. Wu, J. Luan, X. Tan, and L. Zhou, "Xiaoicesing: A high-quality and integrated singing voice synthesis system," in *21st Annual Conference of the International Speech Communication Association, Interspeech 2020*, 2020, pp. 1306–1310.

[31] H. J. Park, S. W. Yang, J. S. Kim, W. Shin, and S. W. Han, "Triaan-vc: Triple adaptive attention normalization for any-to-any voice conversion," in *IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2023*, 2023, pp. 1–5.