X-VC: Zero-shot Streaming Voice Conversion in Codec Space

Qixi Zheng
Shanghai Jiao Tong University
Shanghai, China
jerrister.150329@sjtu.edu.cn

Yuxiang Zhao
Shanghai Jiao Tong University
Shanghai, China

Tianrui Wang
Tianjin University
Tianjin, China

Wenxi Chen
Shanghai Jiao Tong University,
Shanghai Innovation Institute
Shanghai, China

Kele Xu
State Key Laboratory of Complex &
Critical Software Environment
Changsha, China

Yikang Li
Shanghai Innovation Institute
Shanghai, China

Qinyuan Cheng
Fudan University,
Shanghai Innovation Institute
Shanghai, China

Xipeng Qiu
Fudan University,
Shanghai Innovation Institute
Shanghai, China

Kai Yu
Shanghai Jiao Tong University
Shanghai, China

Xie Chen*
Shanghai Jiao Tong University,
Shanghai Innovation Institute
Shanghai, China
chenxie95@sjtu.edu.cn

arXiv:2604.12456v3 [eess.AS] 4 Sep 2026

Abstract

Zero-shot voice conversion (VC) aims to convert a source utterance into the voice of an unseen target speaker while preserving its linguistic content. Although recent systems have improved conversion quality, building zero-shot VC systems for interactive scenarios remains challenging because high-fidelity speaker transfer and low-latency streaming inference are difficult to achieve simultaneously. In this work, we present X-VC, a zero-shot streaming VC system that performs one-step conversion in the latent space of a pretrained neural codec. X-VC uses a dual-conditioning acoustic converter that jointly models source codec latents and frame-level acoustic conditions derived from target reference speech, while injecting utterance-level target speaker information through adaptive normalization. To reduce the mismatch between training and inference, we train the model with generated paired data and a role-assignment strategy that combines standard, reconstruction, and reversed modes. For streaming inference, we further adopt a chunkwise inference scheme with overlap smoothing that is aligned with the segment-based training paradigm of the codec. Experiments on Seed-TTS-Eval show that X-VC achieves the best streaming WER in both English and Chinese, strong speaker similarity in same-language and cross-lingual settings, and substantially lower offline real-time factor than the compared baselines. These results suggest that codec-space one-step conversion is a practical approach for building high-quality low-latency zero-shot VC systems. Our audio samples, code and checkpoints are released at https://github.com/Jerrister/X-VC.

CCS Concepts

• **Computing methodologies** → **Natural language generation**.

Keywords

voice conversion, streaming voice conversion, speech codec, speech generation

*Corresponding author.

1 Introduction

Voice conversion (VC) aims to transform a source utterance so that it sounds as if spoken by a target speaker while preserving the original linguistic content. It has broad applications in multimedia and speech technologies, including dubbing and localization, character voice editing in films, games, and animation, personalized speech generation, and assistive communication [23, 33]. Zero-shot or any-to-any VC further requires the model to generalize to unseen speakers without speaker-specific fine-tuning. A practical zero-shot VC system therefore needs to preserve source linguistic content, accurately transfer target speaker characteristics, and support low-latency streaming inference in a unified framework.

Prior work often approaches zero-shot VC via an analysis and synthesis pipeline, extracting speaker-independent content from the source and combining it with target information to reconstruct the waveform [18–20, 28, 35, 36]. While these studies demonstrate the feasibility of non-parallel zero-shot VC, recent systems highlight the need for richer target-side conditioning. For example, SEF-VC learns timbre directly from reference speech via cross-attention rather than explicit embeddings [17], and models like Vevo and vec2wav 2.0 show that timbre-dependent target conditioning substantially improves conversion quality [12, 42]. Consequently, fine-grained target timbre transfer remains a central challenge in zero-shot VC, especially for unseen speakers.

To better align training supervision with the inference-time conversion scenario, recent work has also begun to leverage generated paired data. Instead of relying only on self- or cross-reconstruction

from real utterances, this paradigm constructs pseudo-parallel supervision in which the linguistic content comes from one utterance while the target timbre is provided by another. A representative example is Seed-VC [21], which explicitly identifies timbre leakage, insufficient timbre representation, and training–inference inconsistency as key obstacles in zero-shot VC. By introducing content-aligned generated pairs and conditioning on the full reference speech context, Seed-VC shows that generated-data training can provide stronger supervision for learning content–timbre recombination in zero-shot conversion. This suggests that the key issue is not whether generated data should replace conventional zero-shot VC pipelines, but how target-side information can be more effectively exploited for voice conversion.

At the same time, many real-world scenarios require low-latency or real-time conversion, which places stringent constraints on model design. Recent streaming VC systems have shown that chunkwise or blockwise processing is a viable route to low-latency inference [22, 24, 25, 37, 38, 43]. However, maintaining strong speaker similarity, content fidelity, and naturalness under streaming constraints remains challenging, especially in zero-shot settings where the target speaker is unseen and only reference speech is available. As a result, jointly achieving high-fidelity target timbre transfer, robust content preservation, and low-latency streaming remains a central open problem in practical zero-shot VC.

Another relevant observation comes from recent speech generation research, where codec tokens and codec latents have become increasingly attractive modeling spaces. Modern neural codecs provide high-fidelity reconstruction and compact intermediate representations [8, 39], while codec-based generation has driven rapid progress in zero-shot speech generation [2, 15, 16, 34]. This suggests that codec space is a natural option for VC as well: the conversion model can focus on transforming a reconstruction-oriented latent representation, while waveform synthesis is delegated to a pretrained codec decoder. Therefore, the latent space of a pretrained neural codec provides a powerful and efficient interface for high-quality, low-latency voice conversion.

Yet operating in codec space does not remove the central conditioning challenge. High-quality VC requires both frame-level acoustic cues and utterance-level speaker information. Fine-grained target timbre is better captured by sequence-level or full-reference conditioning [12, 17, 20, 21], while utterance-level speaker representations remain important for global identity consistency [3, 18, 29]. In our setting, these signals are heterogeneous: source speech is represented in codec latent space, whereas the frame-level condition is represented in mel space. A central difficulty is therefore how to model fine-grained frame-level target cues and utterance-level speaker information under streaming constraints. More broadly, conditioning mechanisms have evolved from feature modulation schemes such as AdaIN and FiLM [14, 27] to transformer-based conditional generation architectures. Recent speech generation systems such as E2 TTS and F5-TTS show that DiT-style backbones can be effectively adapted to speech generation [6, 9]. For heterogeneous inputs, architectures such as MMDiT provide a more relevant reference, where different inputs are first represented separately and then allowed to interact through joint attention [10, 26]. These developments suggest that the key challenge is to preserve the structural differences between conditions while enabling effective interaction among them.

To address these challenges, while drawing on the above insights, we present **X-VC**, a zero-shot streaming voice conversion system in codec space. X-VC performs one-step conversion in the latent space of a pretrained neural codec. A dual-conditioning acoustic converter is introduced to jointly model source codec latents and reference mel-based frame-level conditions, while utterance-level speaker embeddings are injected through adaptive normalization. High-quality generated pairs are used as supervision during training, and chunkwise inference is applied in streaming scenarios to enable low-latency conversion. The main contributions of this work are summarized as follows:

- We present a codec-space framework for zero-shot streaming voice conversion, where conversion is performed in the codec latent space rather than directly in waveform or spectrogram space.
- We design a dual-conditioning acoustic converter that jointly models frame-level acoustic conditions and utterance-level speaker information for codec-space latent transformation.
- We develop a practical training and inference framework, including generated-pair training with flexible role assignment and chunkwise streaming inference aligned with codec segmentation, improving robustness and enabling low-latency conversion, while naturally supporting cross-lingual voice conversion.
- We validate the proposed approach on English, Chinese, and cross-lingual settings, and show through extensive experiments and ablations the importance of dual-condition modeling and the proposed training strategy.

2 Related Work

2.1 High-Fidelity Zero-Shot VC

Recent zero-shot VC systems increasingly explore richer target-side information to improve speaker similarity and naturalness. SEF-VC [17] learns speaker timbre directly from reference speech through cross-attention, avoiding explicit speaker embeddings. Vevo [42] studies controllable zero-shot voice imitation with self-supervised disentanglement, while vec2wav 2.0 [12] combines discrete content tokens with timbre-dependent features to improve conversion quality. Compared with earlier zero-shot VC systems, these methods place more emphasis on how target-side information is represented and injected into the model. However, retaining such rich target-side conditioning in low-latency streaming zero-shot VC remains underexplored.

2.2 VC with Pseudo-Parallel Supervision

Seed-VC [21] represents a different direction, where generated paired data are introduced to better match the inference-time conversion scenario. It constructs pseudo-parallel pairs in which the linguistic content is inherited from one utterance while the target timbre is provided by another, and trains a diffusion-based conversion model with full-reference conditioning. This line of work demonstrates the value of generated supervision for zero-shot VC, while how to further exploit generated pairs during training has not been explicitly explored.

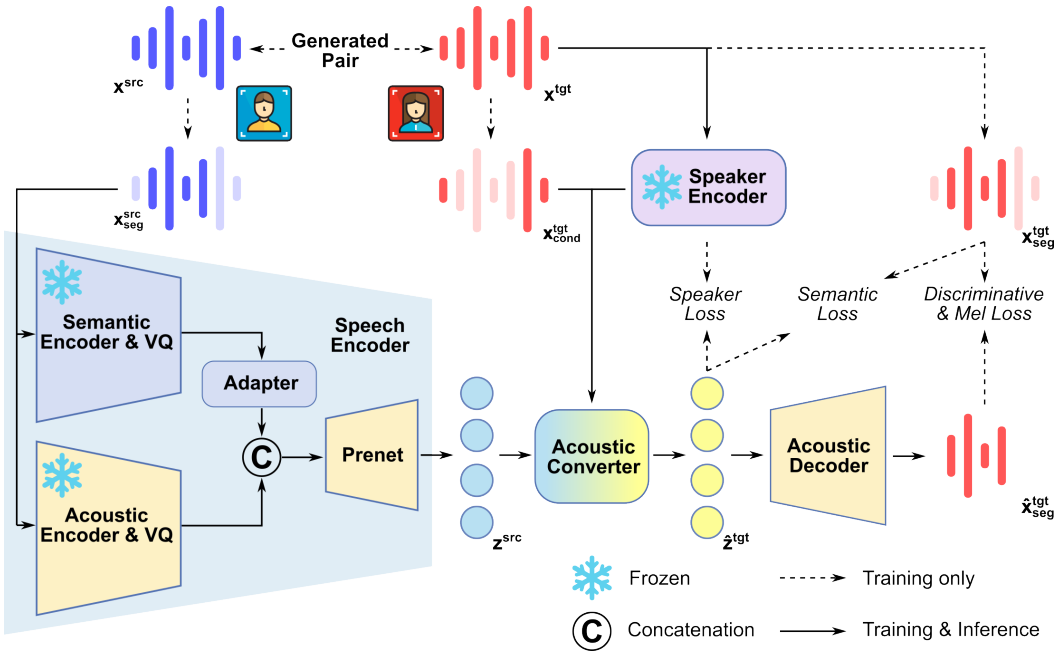


Figure 1: Overall framework of X-VC. A pretrained speech encoder maps the input waveform into latent representations, which are then transformed by the acoustic converter conditioned on target-side reference speech and finally decoded into waveform.

2.3 Dual-Condition Modeling

Voice conversion often requires the joint use of frame-level acoustic information and utterance-level speaker characteristics [28]. These two forms of conditioning are complementary, providing both time-varying and global target cues, but their differences in structure and granularity make them non-trivial to integrate within a unified model [18]. To handle such heterogeneous conditions, models like the MMDiT [10] use separate projections for different modalities and enable interaction through joint attention, while adaptive layer normalization (AdaLN) [26] provides an effective way to inject global conditioning signals through feature modulation. These developments suggest a promising direction for VC: jointly modeling frame-level and utterance-level conditions while preserving their structural differences.

2.4 Streaming Voice Conversion

Streaming VC focuses on reducing inference latency while preserving linguistic content and target speaker identity. StreamVC [38] adopts chunkwise processing for real-time low-latency conversion, Conan [43] develops a chunkwise online zero-shot adaptive VC framework, and MeanVC [22] introduces mean flows for single-step streaming zero-shot VC. Nevertheless, maintaining strong target speaker similarity and content fidelity in zero-shot settings remains challenging for streaming models.

3 Methodology

3.1 Overview

We formulate zero-shot voice conversion in the latent space of a pretrained neural audio codec. Given a source utterance $\mathbf{x}^{\text{src}}$ and a target-side reference utterance $\mathbf{x}^{\text{tgt}}$, we first extract a short segment $\mathbf{x}^{\text{src}}_{\text{seg}}$ and encode it into codec latent representations using a pretrained speech encoder:

$$\mathbf{z} = \mathcal{E}(\mathbf{x}^{\text{src}}_{\text{seg}}), \tag{1}$$

where $\mathbf{z}$ denotes the source codec latent sequence.

For conditioning, we construct a target-side reference signal by removing the corresponding segment from the target utterance, denoted as $\mathbf{x}^{\text{tgt}}_{\text{cond}}$. From this signal, we extract two complementary types of information: a frame-level acoustic condition (e.g., mel-spectrogram) and an utterance-level speaker representation:

$$\mathbf{c} = \mathcal{F}(\mathbf{x}^{\text{tgt}}_{\text{cond}}), \quad \mathbf{g} = \mathcal{S}(\mathbf{x}^{\text{tgt}}_{\text{cond}}), \tag{2}$$

where $\mathcal{F}(\cdot)$ denotes the acoustic feature extractor and $\mathcal{S}(\cdot)$ denotes the speaker encoder.

The acoustic converter performs voice conversion directly in the codec latent space:

$$\hat{\mathbf{z}} = f(\mathbf{z}, \mathbf{c}, \mathbf{g}), \tag{3}$$

where $\hat{\mathbf{z}}$ is the converted codec latent sequence. Finally, the codec decoder reconstructs the waveform:

$$\hat{\mathbf{x}} = \mathcal{D}(\hat{\mathbf{z}}). \tag{4}$$

As illustrated in Figure 1, we first encode the input waveform into codec latent representations using a pretrained speech encoder. Voice conversion is then performed on these latent representations by the acoustic converter, before the converted latents are decoded back into waveform. This formulation allows us to cast voice conversion as a transformation in a pretrained codec latent space, rather than directly generating waveform samples.

3.2 Codec Instantiation with SAC

In this work, the abstract speech encoder introduced in Section 3.1 is instantiated by the frontend of the pretrained SAC codec [5]. Concretely, it includes the semantic encoder, VQ and adapter, the acoustic encoder and VQ, and the prenet.

For the zero-shot streaming voice conversion task considered in this work, SAC offers two particularly useful properties.

First, SAC provides a strong latent frontend for high-quality speech reconstruction. Its dual-stream design combines semantic and acoustic information, yielding latent representations that preserve both linguistic content and fine-grained acoustic characteristics. This makes SAC a suitable backbone for voice conversion, where the model must preserve source content while adapting speaker-related acoustic traits.

Second, SAC is trained on short speech segments, which makes it naturally compatible with chunkwise processing at inference time. Although the codec itself is not strictly causal, this segment-based training paradigm is favorable for low-latency chunkwise conversion with overlap smoothing, as described in Section 3.5.

At the same time, from the perspective of voice conversion, we do not regard SAC as providing perfectly disentangled semantic and acoustic representations. Although SAC organizes speech information into semantic and acoustic pathways, its acoustic branch is still optimized for accurate waveform reconstruction and therefore may retain content-related information in addition to speaker and acoustic traits. In this sense, SAC is better viewed as a high-quality codec frontend than as a fully disentangled tokenizer. This is also different from works that explicitly strengthen semantic–acoustic separation through dedicated supervision and recombination-style training, such as DSA-Tokenizer [41].

For this reason, we do not perform conversion directly on the acoustic codes of a single pathway. Instead, we perform conversion after the prenet, where information from both pathways has already been integrated into a unified latent space. This stage corresponds to the output of the speech encoder in Figure 1. Operating on this unified representation allows the converter to jointly model the content that should be preserved and the acoustic characteristics that should be transformed, which is better aligned with the goal of voice conversion.

We build upon the pretrained codec without redesigning its architecture. During training, the codec encoder-side components are kept frozen so that the latent space remains stable and learning is focused on the conversion module. In practice, this includes the semantic and acoustic encoding pathways, together with their quantization modules. The proposed acoustic converter (detailed next in Section 3.3) is then optimized to transform the source codec latents conditioned on the target-side reference speech, while the decoder reconstructs the converted waveform from the transformed latents.

3.3 Acoustic Converter

The acoustic converter is the core component of the proposed system. It transforms source codec latents toward the target speaker characteristics while preserving the linguistic content. An overview of its architecture is shown in Figure 2.

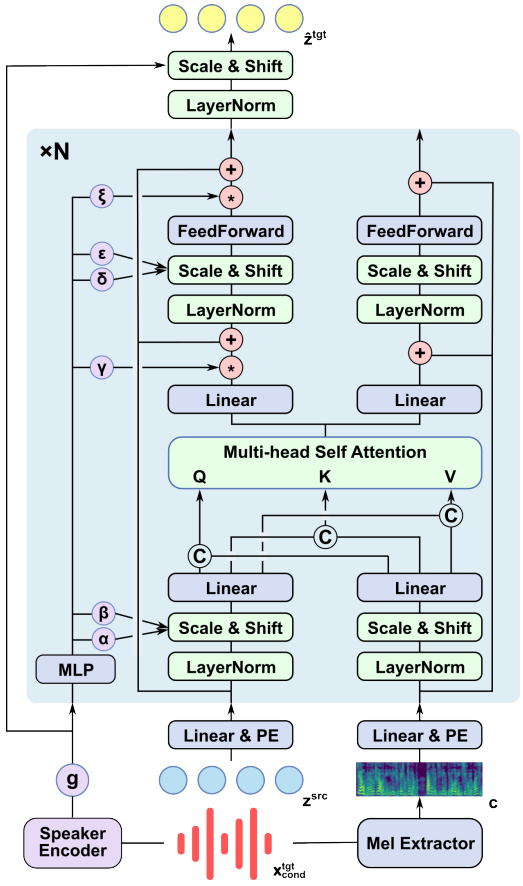


Figure 2: Architecture of the acoustic converter. Source codec latents and the mel sequence extracted from x^{tgt}_{cond} are jointly updated through shared attention, while the utterance-level speaker embedding modulates each block through adaptive normalization.

3.3.1 Design Overview. Our goal is to incorporate two complementary forms of conditioning into the conversion process: a frame-level acoustic condition and an utterance-level speaker representation. The former provides detailed, time-varying acoustic patterns of the target speaker, while the latter provides a stable global description of speaker identity.

A key challenge is that these signals are defined in different representation spaces. The source is represented in codec latent space, while the frame-level condition is derived from mel-spectrograms. These representations cannot be directly combined at the input level. To address this, we design the converter as a two-branch architecture with separate input projections for the source and the condition. Inspired by recent multimodal transformer [10] architectures, this design allows heterogeneous inputs to be processed independently before enabling interaction between them.

During training, the mel sequence fed to the conditioning branch is extracted from x^{tgt}_{cond}, i.e., the target utterance after removing the selected target segment. This prevents trivial copying and keeps the

conditioning setup consistent with the training pipeline described in Section 3.4.

3.3.2 Interaction Between Source and Condition. After projection, the source and condition sequences are processed jointly through a stack of transformer blocks. In each block, the two sequences are concatenated for the attention operation, enabling information exchange between them.

Importantly, the condition is not treated as a fixed context. Both the source and condition representations are updated across layers. This allows the model to progressively refine both representations, rather than only conditioning the source on a static reference.

In the context of voice conversion, this behavior is particularly important. The source latent carries the linguistic content to be preserved, while the acoustic condition provides detailed spectral patterns of the target speaker. By jointly updating both representations, the model can progressively align the source content with the target acoustic characteristics.

3.3.3 Global Speaker Conditioning. In addition to the frame-level acoustic condition, we incorporate an utterance-level speaker embedding as a global conditioning signal. This embedding is injected into the network through adaptive normalization layers, which modulate the hidden representations across layers.

The global speaker representation complements the frame-level condition. While the mel-spectrogram provides local acoustic details, the speaker embedding provides a consistent identity signal over the entire utterance. This helps stabilize the conversion and improves speaker consistency.

3.4 Training Strategy

Our training strategy is based on paired data with matched linguistic content and different speaker characteristics, together with flexible role assignment between source and target. An overview is shown in Figure 3.

3.4.1 Data Construction. For each randomly paired utterance pair $(\mathbf{x}_0, \mathbf{x}_1)$, generated samples are constructed in both directions, i.e., $\mathbf{x}_{0\to1}$ and $\mathbf{x}_{1\to0}$. Each generated sample preserves the linguistic content of one utterance while adopting the speaker characteristics of the other. This process is performed using a pretrained offline voice conversion model.

As a result, each training pair consists of two utterances with matched linguistic content but different speaker characteristics, one real and one generated. Such pairs provide suitable supervision for voice conversion, where content and speaker identity are disentangled at the data level.

During training, both source and target waveforms are cropped into short segments. For the frame-level conditioning branch, we remove the selected target segment from the original target utterance and use the remaining part, denoted as $\mathbf{x}^{\text{tgt}}_{\text{cond}}$, to extract the mel-spectrogram condition. This prevents the conditioning branch from directly accessing the target segment and avoids trivial copying.

3.4.2 Role Assignment. Given a paired example consisting of two utterances with the same content but different speaker characteristics, we consider different assignments of source and target roles.

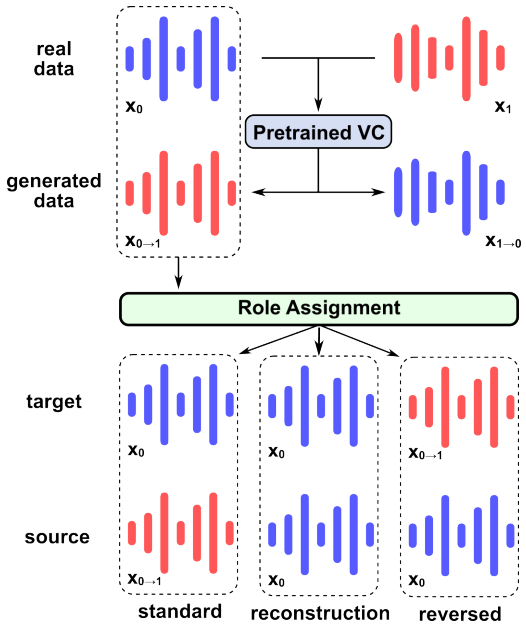


Figure 3: Training data construction and role assignment. A pretrained VC model generates paired utterances from real speech, which are then used to form source-target pairs under standard, reconstruction, and reversed modes.

A standard choice is to use the generated utterance as the source and the corresponding real utterance as the target. For example, given the pair $(\mathbf{x}_{0\to1}, \mathbf{x}_0)$, the standard assignment uses $\mathbf{x}_{0\to1}$ as the source and $\mathbf{x}_0$ as the target.

In addition, we also use the reversed assignment, where the real utterance is used as the source and the generated utterance is used as the target. For the same example, the reversed assignment uses $\mathbf{x}_0$ as the source and $\mathbf{x}_{0\to1}$ as the target.

We further include a self-reconstruction assignment, where the source and target are taken from the same real utterance, e.g., $(\mathbf{x}_0, \mathbf{x}_0)$.

During training, each paired example is assigned to one of the three modes—standard, self-reconstruction, or reversed—by random sampling according to probabilities p_{std}, p_{recon}, and p_{rev}.

These assignments do not change the semantic correspondence of the pair, but expose the model to a more diverse input-output distribution. In particular, both the encoder and decoder observe real and generated speech during training, which improves robustness and reduces distribution mismatch.

3.4.3 Training Objectives. The training objective follows the original SAC formulation, with the main difference being the definition of the prediction target.

Given a selected target segment under the role assignment above, the model is trained to reconstruct this target from the corresponding source and conditioning signals. Therefore, all losses are computed with respect to the chosen target segment.

We use the same loss terms as in SAC, including the semantic MSE loss, mel reconstruction loss, speaker similarity MSE loss,

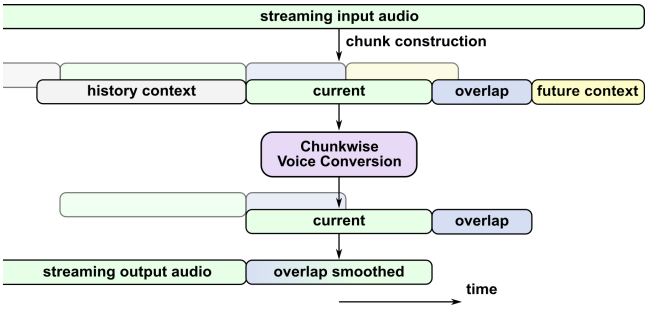


Figure 4: Chunkwise streaming inference of X-VC. Each step uses a fixed processing window composed of history, current, overlap, and future regions. Only the current region is emitted as valid output, while the front part of the current chunk is blended with the overlap retained from the previous step to ensure continuity.

and adversarial discriminator loss. Since the vector quantization modules are frozen in our setting, the VQ-related loss is removed.

3.5 Chunkwise Streaming Inference

Although the underlying codec is not strictly causal, its segment-based training paradigm makes it amenable to chunkwise inference. To enable low-latency voice conversion, chunkwise inference is applied only in streaming scenarios, as illustrated in Figure 4.

During streaming inference, the conditioning signals are precomputed from the reference speech. Specifically, the frame-level acoustic features and the utterance-level speaker embedding are extracted once before streaming begins, and are reused for all subsequent chunks. Therefore, only the source speech is processed in a streaming manner. The input speech is processed in consecutive chunks. At each step, a fixed-length window is constructed, consisting of four parts: a history region, a current region, an overlap region for boundary smoothing, and an optional future context. The total window length is aligned with the training segment, ensuring consistency between training and inference. The entire window is fed into the codec and acoustic converter, while only the current region is retained as valid output. The surrounding context is introduced to improve local continuity but is not directly emitted. This design allows the model to leverage sufficient temporal context while maintaining low latency.

To ensure smooth transitions across chunk boundaries, overlap smoothing is applied. Specifically, the overlap retained from the previous step is blended with the front part of the current chunk using a cosine cross-fade, reducing discontinuities in the reconstructed waveform.

The algorithmic latency is determined by both the required input context and the computation time. We decompose it as:

$$T_{\text{latency}} = T_{\text{model}} + T_{\text{compute}}, \tag{5}$$

where T_{model} denotes the amount of input context required before producing valid output, and T_{compute} denotes the runtime of the model. In this system,

$$T_{\text{model}} = T_{\text{current}} + T_{\text{overlap}} + T_{\text{future}}, \tag{6}$$

$$T_{\text{compute}} = T_{\text{enc}} + T_{\text{convert}} + T_{\text{dec}}. \tag{7}$$

Here, T_{compute} includes all online processing steps in the pipeline, including encoding, acoustic conversion, and decoding, corresponding to an end-to-end definition of streaming inference. Conditioning features derived from the reference speech are computed beforehand and thus excluded. Overall, this chunkwise strategy approximates streaming behavior and maintains low latency without requiring a fully causal model, all while preserving the reconstruction quality of the pretrained codec.

4 Experimental Setup

4.1 Model Configuration

4.1.1 Codec, Feature Extraction, and Speaker Encoder. A pretrained SAC codec [5] is adopted, using its 16 kHz 62.5 Hz configuration. The speech encoder encodes speech into codec latent representations with a dimensionality of 1024. For frame-level conditioning, mel-spectrograms are extracted from the reference speech using a standard mel extractor with 128 mel bins, which defines the sequence-level conditioning dimension. For utterance-level conditioning, a pretrained speaker encoder based on ERes2Net [7] is used to produce 192-dimensional speaker embeddings.

4.1.2 Converter. The acoustic converter follows the architecture described in Section 3.3. It is implemented as a stack of transformer blocks that jointly model source codec latents and frame-level acoustic conditions. The input dimensionalities are determined by the upstream components: the codec latent has 1024 channels, the frame-level condition has 128 channels (mel-spectrogram), and the utterance-level speaker embedding has dimension 192. A hidden dimension of 512 is used, with 6 transformer layers and 8 attention heads, where each head has a dimension of 64. A feed-forward expansion ratio of 4 is adopted in each block.

4.1.3 Training Details. Training is performed using the strategy described in Section 3.4. $(p_{\text{std}},\ p_{\text{recon}},\ p_{\text{rev}})$ are set to $(0.4, 0.2, 0.4)$. The model is trained for 446k steps on 8 NVIDIA H200 GPUs, with a per-GPU batch size of 24. The AdamW optimizer is used with a learning rate of 1×10^{-4} and $\beta = (0.8, 0.9)$. An exponential learning rate scheduler is applied with a decay factor of 0.999996 and a minimum learning rate of 10^{-6}. Gradient clipping is applied with a maximum norm of 5, and exponential moving average (EMA) is used to stabilize training.

4.2 Datasets

4.2.1 Training Dataset. The model is trained on a mixture of the Emilia dataset [13] and the LibriTTS dataset [40]. For the Emilia dataset, data filtering is applied based on DNSMOS [31], and only samples with DNSMOS scores higher than 3.45 are retained to ensure data quality. In total, approximately 10,000 hours of English and Chinese speech data are collected. Based on these data, additional paired samples are generated using a pretrained Seed-VC$_{small}$ model, following the procedure described in Section 3.4.1. This results in approximately 20,000 hours of generated data. Following SAC, each audio sample is resampled to 16 kHz and randomly cropped into 2.4-second segments during training.

Table 1: Streaming zero-shot voice conversion results on Seed-TTS-Eval. Best values are shown in bold. ↑ and ↓ indicate higher or lower values are better. Because the test set is non-parallel, only GT WER is reported, using the source utterances as content references; GT SIM and GT UTMOS are omitted as they are not meaningful in this setting.

Model	#Param.		Seed-TTS *test-en*			Seed-TTS *test-zh*			Latency (ms) ↓	
	Total	Converter	WER (%) ↓	SIM ↑	UTMOS ↑	WER (%) ↓	SIM ↑	UTMOS ↑	T_{model}	T_{compute}
Ground Truth	-	-	1.96	-	-	1.33	-	-	-	-
Seed-VC$_{tiny}$	262M	27M	3.31	0.40	2.97	3.36	0.60	**2.46**	380	120.75
MeanVC	47M	14M	-	-	-	4.89	**0.72**	2.22	250	**32.12**
X-VC (Ours)	539M	44M	**3.14**	**0.62**	**3.07**	**2.65**	**0.72**	2.35	240	58.17

Table 2: Subjective speaker similarity (SMOS ↑) with 95% confidence intervals computed from the t-test.

Model	Seed-TTS *test-en*	Seed-TTS *test-zh*
Seed-VC	3.56 ± 0.12	3.38 ± 0.17
MeanVC	-	3.44 ± 0.15
X-VC (Ours)	**3.98 ± 0.10**	**3.89 ± 0.13**

4.2.2 Evaluation Dataset. We evaluate our model on the Seed-TTS-Eval benchmark [1], which offers non-parallel pairs for zero-shot voice conversion. The evaluation includes English (*test-en*) and Chinese (*test-zh*) settings. In addition, we construct a cross-lingual evaluation setting based on the Seed-TTS-Eval dataset. Specifically, we select the first 1,000 utterances from the English and Chinese subsets, and form cross-lingual pairs by exchanging the source and reference utterances across languages. This results in two subsets: English-to-Chinese and Chinese-to-English, which evaluate the ability of the model to transfer speaker characteristics across languages while preserving the original linguistic content.

4.3 Baseline Models

We compare the proposed method with representative zero-shot voice conversion models trained on the Emilia dataset.

Seed-VC [21] is a diffusion-based voice conversion framework. Following the guidelines in its official repository[1], we evaluate Seed-VC$_{\text{small}}$ in the offline setting and Seed-VC$_{\text{tiny}}$ in both streaming and offline settings. MeanVC [22] is a flow-based voice conversion model built upon the MeanFlow framework. We use its released 200 ms chunk version[2] for both streaming and offline settings. We note that its released offline inference script is also chunkwise rather than full-utterance, which should be taken into account when interpreting its offline RTF. Also, MeanVC is omitted from the English-source evaluations as its content encoder module only supports Chinese.

4.4 Evaluation Metrics

We evaluate performance across both generation quality and inference efficiency. To assess speech quality, we use Word Error Rate (WER) to measure linguistic preservation, employing Whisper-large-v3 [30] for English and Paraformer-zh [11] for Chinese. Speaker identity consistency is evaluated via Speaker Similarity (SIM), computed as the cosine similarity between speaker embeddings extracted by a WavLM-based model [4], while UTMOS [32] is utilized to objectively estimate speech naturalness without human annotations. For inference efficiency, all metrics are measured end-to-end on a single NVIDIA RTX 3090 GPU. Streaming latency is decomposed into model-induced latency (T_{model}), determined by the required current, overlap, and future contexts, and computation time (T_{compute}), which represents the average runtime per chunk including encoding, acoustic conversion, and decoding. Finally, for offline inference, we report the real-time factor (RTF) as the ratio of the total processing time to the input audio duration.

In addition to objective metrics, we perform a subjective evaluation to assess speaker similarity. We conduct a speaker similarity mean opinion score (SMOS) test with 15 listeners, who are asked to rate the speaker similarity between the converted speech and the target reference on a 5-point scale (1: completely different, 5: identical speaker). All samples are presented in random order to avoid bias.

5 Results and Discussion

5.1 Streaming Performance

Table 1 reports the streaming zero-shot voice conversion results on Seed-TTS-Eval. Overall, X-VC provides the strongest streaming performance among the compared systems. It achieves the best WER on both *test-en* and *test-zh*, indicating more reliable content preservation under online constraints. At the same time, its SIM is the best on English and tied for the best on Chinese, while its UTMOS remains competitive and is the best on English. Taken together, these results suggest that X-VC offers a better overall balance of intelligibility, speaker similarity, and perceptual quality than the streaming baselines.

In addition to objective metrics, the subjective speaker similarity results in Table 2 are also examined. Results show that X-VC achieves higher SMOS scores compared to the baseline systems, indicating better perceptual speaker similarity. Meanwhile, X-VC exhibits lower variance in listener ratings, suggesting more consistent conversion quality across samples. These observations are consistent with the objective SIM results.

In terms of latency, X-VC has a model-induced latency T_{model} of 240 ms, with 120 ms for T_{current}, 20 ms for T_{overlap}, and 100 ms

[1] https://github.com/Plachtaa/seed-vc
[2] https://github.com/ASLP-lab/MeanVC

Table 3: Offline zero-shot voice conversion results on Seed-TTS-Eval.

Model	#Param.		Seed-TTS *test-en*			Seed-TTS *test-zh*			RTF ↓
	Total	Converter	WER (%) ↓	SIM ↑	UTMOS ↑	WER (%) ↓	SIM ↑	UTMOS ↑	
Ground Truth	-	-	1.96	-	-	1.33	-	-	-
Seed-VC$_{small}$	296M	98M	2.57	0.56	3.37	2.52	**0.73**	2.73	0.161
Seed-VC$_{tiny}$	262M	27M	**2.24**	0.41	**3.67**	**1.79**	0.60	**3.08**	0.069
MeanVC	47M	14M	-	-	-	3.89	**0.73**	2.81	0.094
X-VC (Ours)	539M	44M	2.83	**0.63**	3.31	1.99	**0.73**	2.69	**0.014**

Table 4: Offline cross-lingual results on Seed-TTS-Eval.

Model	English-to-Chinese		Chinese-to-English	
	WER (%)↓	SIM ↑	WER (%)↓	SIM ↑
Ground Truth	1.91	-	1.14	-
Seed-VC$_{small}$	2.82	**0.52**	2.36	0.42
Seed-VC$_{tiny}$	**2.23**	0.34	**1.70**	0.29
MeanVC	-	-	3.89	0.42
X-VC (Ours)	2.67	**0.52**	2.15	**0.49**

Table 5: Offline ablation results on Seed-TTS *test-zh*. All models are trained for 300k steps for fair comparison.

Model	WER (%)↓	SIM ↑
X-VC	**2.02**	**0.72**
Condition Modeling		
w/o updating frame-level condition **c**	2.15	0.66
w/o utterance-level condition **g**	2.20	0.61
Data Construction		
standard only ($p_{\text{std}} = 1.0$)	2.31	**0.72**
reversed only ($p_{\text{rev}} = 1.0$)	2.14	0.71
w/o reconstruction ($p_{\text{std}} = p_{\text{rev}} = 0.5$)	2.14	**0.72**

for T_{future}. Its computation latency T_{compute} is higher than that of MeanVC, but remains lower than that of Seed-VC$_{tiny}$. This comparison should be interpreted together with the streaming context design. To be consistent with the 2.4-second segment setting used in training, each streaming step of X-VC uses a 2.4-second processing window, which is substantially larger than those of the compared baselines, especially in terms of history context. This larger window also provides stronger contextual support for chunk-wise streaming inference, leading to more stable conversion across chunk boundaries. Even with this larger context, X-VC remains streamable, indicating that the proposed one-step codec-space conversion is computationally practical for low-latency inference.

5.2 Offline Performance

Table 3 shows the offline zero-shot voice conversion results. Under the offline setting, X-VC remains competitive in quality while exhibiting a clear advantage in inference efficiency. In particular, it achieves the best SIM on English and tied-best SIM on Chinese, while its WER remains competitive, especially on Chinese. Its UTMOS is not always the best, but stays in the same overall range as the strongest baselines. These results indicate that X-VC is particularly effective at speaker similarity preservation while maintaining solid intelligibility and naturalness.

A notable advantage of X-VC is efficiency. It achieves an RTF of 0.014, which is substantially lower than all compared baselines. This is consistent with the one-step nature of the proposed method: once the codec latents are extracted, conversion is performed directly in latent space without iterative generation. We also note that the released offline script of MeanVC performs chunkwise inference rather than full-utterance inference, which partly explains its relatively low RTF. Nevertheless, X-VC remains considerably faster. Overall, the offline results show that X-VC provides a strong quality–efficiency trade-off.

5.3 Cross-lingual Evaluation

Table 4 presents the cross-lingual results. X-VC remains competitive in both language-mismatched settings. For English-to-Chinese conversion, it matches the best SIM and achieves WER close to the best baseline. For Chinese-to-English conversion, it achieves the best SIM while maintaining competitive WER. These results are consistent with the same-language experiments: X-VC is particularly strong at preserving target speaker characteristics, while remaining robust in content preservation under cross-lingual mismatch.

5.4 Ablation Studies

Table 5 reports two groups of ablations on Seed-TTS *test-zh*.

For condition modeling, removing either the utterance-level condition **g** or the updating of frame-level condition **c** degrades both WER and SIM, indicating that the two conditioning signals are complementary and both important for content preservation and speaker similarity.

For data construction, the ablations mainly affect WER while leaving SIM largely unchanged. This suggests that the multi-mode role-assignment strategy primarily improves robustness and intelligibility, with more limited impact on speaker similarity.

Overall, the ablations confirm the importance of both the dual-conditioning converter and the proposed data-construction strategy.

6 Conclusion

This paper presented X-VC, a zero-shot streaming voice conversion system that performs one-step conversion in the latent space of the pretrained SAC codec. The proposed system combines a dual-conditioning acoustic converter, generated-pair training with flexible role assignment, and chunkwise streaming inference with overlap smoothing. Experiments on Seed-TTS-Eval demonstrate that X-VC achieves the best streaming WER in both English and Chinese, strong speaker similarity in both same-language and cross-lingual settings, and substantially lower offline RTF than the compared baselines. Ablation studies further confirm the importance of jointly updating the frame-level condition stream, incorporating utterance-level global speaker conditioning, and using role assignment during training.

Overall, the results suggest that codec-space one-step conversion is a promising direction for practical zero-shot voice conversion, especially when both quality and latency are important.

References

[1] Philip Anastassiou, Jiawei Chen, Jitong Chen, Yuanzhe Chen, Zhuo Chen, Ziyi Chen, Jian Cong, Lelai Deng, Chuang Ding, Lu Gao, Mingqing Gong, Peisong Huang, Qingqing Huang, Zhiying Huang, Yuanyuan Huo, Dongya Jia, Chumin Li, Feiya Li, Hui Li, Jiaxin Li, Xiaoyang Li, Xingxing Li, Lin Liu, Shouda Liu, Sichao Liu, Xudong Liu, Yuchen Liu, Zhengxi Liu, Lu Lu, Junjie Pan, Xin Wang, Yuping Wang, Yuxuan Wang, Zhen Wei, Jian Wu, Chao Yao, Yifeng Yang, Yuanhao Yi, Junteng Zhang, Qidi Zhang, Shuo Zhang, Wenjie Zhang, Yang Zhang, Zilin Zhao, Dejian Zhong, and Xiaobin Zhuang. 2024. Seed-TTS: A Family of High-Quality Versatile Speech Generation Models. arXiv:2406.02430 [eess.AS] https://arxiv.org/abs/2406.02430

[2] Zalán Borsos, Matt Sharifi, Damien Vincent, Eugene Kharitonov, Neil Zeghidour, and Marco Tagliasacchi. 2023. SoundStorm: Efficient Parallel Audio Generation. arXiv:2305.09636 [cs.SD] https://arxiv.org/abs/2305.09636

[3] Edresson Casanova, Julian Weber, Christopher D Shulby, Arnaldo Candido Junior, Eren Gölge, and Moacir A Ponti. 2022. YourTTS: Towards Zero-Shot Multi-Speaker TTS and Zero-Shot Voice Conversion for Everyone. In *Proceedings of the 39th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 162)*, Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (Eds.). PMLR, 2709–2720. https://proceedings.mlr.press/v162/casanova22a.html

[4] Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Jian Wu, Michael Zeng, Xiangzhan Yu, and Furu Wei. 2022. WavLM: Large-Scale Self-Supervised Pre-Training for Full Stack Speech Processing. *IEEE Journal of Selected Topics in Signal Processing* 16, 6 (2022), 1505–1518. doi:10.1109/JSTSP.2022.3188113

[5] Wenxi Chen, Xinsheng Wang, Ruiqi Yan, Yushen Chen, Zhikang Niu, Ziyang Ma, Xiquan Li, Yuzhe Liang, Hanlin Wen, Shunshun Yin, Ming Tao, and Xie Chen. 2025. SAC: Neural Speech Codec with Semantic-Acoustic Dual-Stream Quantization. arXiv:2510.16841 [eess.AS] https://arxiv.org/abs/2510.16841

[6] Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, Jian Zhao, Kai Yu, and Xie Chen. 2025. F5-TTS: A Fairytaler that Fakes Fluent and Faithful Speech with Flow Matching. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (Eds.). Association for Computational Linguistics, Vienna, Austria, 6255–6271. doi:10.18653/v1/2025.acl-long.313

[7] Yafeng Chen, Siqi Zheng, Hui Wang, Luyao Cheng, Qian Chen, and Jiajun Qi. 2023. An Enhanced Res2Net with Local and Global Feature Fusion for Speaker Verification. In *Interspeech 2023*. 2228–2232. doi:10.21437/Interspeech.2023-1294

[8] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. 2023. High Fidelity Neural Audio Compression. *Transactions on Machine Learning Research* (2023). https://openreview.net/forum?id=ivCd8z8zR2 Featured Certification, Reproducibility Certification.

[9] Sefik Emre Eskimez, Xiaofei Wang, Manthan Thakker, Canrun Li, Chung-Hsien Tsai, Zhen Xiao, Hemin Yang, Zirun Zhu, Min Tang, Xu Tan, Yanqing Liu, Sheng Zhao, and Naoyuki Kanda. 2024. E2 TTS: Embarrassingly Easy Fully Non-Autoregressive Zero-Shot TTS. In *2024 IEEE Spoken Language Technology Workshop (SLT)*. 682–689. doi:10.1109/SLT61566.2024.10832320

[10] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. 2024. Scaling Rectified Flow Transformers for High-Resolution Image Synthesis. In *Proceedings of the 41st International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 235)*, Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (Eds.). PMLR, 12606–12633. https://proceedings.mlr.press/v235/esser24a.html

[11] Zhifu Gao, ShiLiang Zhang, Ian McLoughlin, and Zhijie Yan. 2022. Paraformer: Fast and Accurate Parallel Transformer for Non-autoregressive End-to-End Speech Recognition. In *Interspeech 2022*. 2063–2067. doi:10.21437/Interspeech.2022-9996

[12] Yiwei Guo, Zhihan Li, Junjie Li, Chenpeng Du, Hankun Wang, Shuai Wang, Xie Chen, and Kai Yu. 2025. vec2wav 2.0: Advancing Voice Conversion via Discrete Token Vocoders. arXiv:2409.01995 [eess.AS] https://arxiv.org/abs/2409.01995

[13] Haorui He, Zengqiang Shang, Chaoren Wang, Xuyuan Li, Yicheng Gu, Hua Hua, Liwei Liu, Chen Yang, Jiaqi Li, Peiyang Shi, Yuancheng Wang, Kai Chen, Pengyuan Zhang, and Zhizheng Wu. 2024. Emilia: An Extensive, Multilingual, and Diverse Speech Dataset For Large-Scale Speech Generation. In *2024 IEEE Spoken Language Technology Workshop (SLT)*. 885–890. doi:10.1109/SLT61566.2024.10832365

[14] Xun Huang and Serge Belongie. 2017. Arbitrary Style Transfer in Real-Time With Adaptive Instance Normalization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

[15] Zeqian Ju, Yuancheng Wang, Kai Shen, Xu Tan, Detai Xin, Dongchao Yang, Eric Liu, Yichong Leng, Kaitao Song, Siliang Tang, Zhizheng Wu, Tao Qin, Xiangyang Li, Wei Ye, Shikun Zhang, Jiang Bian, Lei He, Jinyu Li, and Sheng Zhao. 2024. NaturalSpeech 3: Zero-Shot Speech Synthesis with Factorized Codec and Diffusion Models. In *Proceedings of the 41st International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 235)*, Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (Eds.). PMLR, 22605–22623. https://proceedings.mlr.press/v235/ju24b.html

[16] Eugene Kharitonov, Damien Vincent, Zalán Borsos, Raphaël Marinier, Sertan Girgin, Olivier Pietquin, Matt Sharifi, Marco Tagliasacchi, and Neil Zeghidour. 2023. Speak, Read and Prompt: High-Fidelity Text-to-Speech with Minimal Supervision. *Transactions of the Association for Computational Linguistics* 11 (12 2023), 1703–1718. doi:10.1162/tacl_a_00618

[17] Junjie Li, Yiwei Guo, Xie Chen, and Kai Yu. 2024. SEF-VC: Speaker Embedding Free Zero-Shot Voice Conversion with Cross Attention. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 12296–12300. doi:10.1109/ICASSP48485.2024.10446160

[18] Jingyi Li, Weiping Tu, and Li Xiao. 2023. Freevc: Towards High-Quality Text-Free One-Shot Voice Conversion. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 1–5. doi:10.1109/ICASSP49357.2023.10095191

[19] Yinghao Aaron Li, Ali Zare, and Nima Mesgarani. 2021. StarGANv2-VC: A Diverse, Unsupervised, Non-Parallel Framework for Natural-Sounding Voice Conversion. In *Interspeech 2021*. 1349–1353. doi:10.21437/Interspeech.2021-319

[20] Yist Y. Lin, Chung-Ming Chien, Jheng-Hao Lin, Hung-yi Lee, and Lin-shan Lee. 2021. 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)*. 5939–5943. doi:10.1109/ICASSP39728.2021.9413699

[21] Songting Liu. 2024. Zero-shot Voice Conversion with Diffusion Transformers. arXiv:2411.09943 [cs.SD] https://arxiv.org/abs/2411.09943

[22] Guobin Ma, Jixun Yao, Ziqian Ning, Yuepeng Jiang, Lingxin Xiong, Lei Xie, and Pengcheng Zhu. 2025. MeanVC: Lightweight and Streaming Zero-Shot Voice Conversion via Mean Flows. arXiv:2510.08392 [eess.AS] https://arxiv.org/abs/2510.08392

[23] Seyed Hamidreza Mohammadi and Alexander Kain. 2017. An overview of voice conversion systems. *Speech Communication* 88 (2017), 65–82. doi:10.1016/j.specom.2017.01.008

[24] Ziqian Ning, Yuepeng Jiang, Pengcheng Zhu, Shuai Wang, Jixun Yao, Lei Xie, and Mengxiao Bi. 2024. Dualvc 2: Dynamic Masked Convolution for Unified Streaming and Non-Streaming Voice Conversion. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 11106–11110. doi:10.1109/ICASSP48485.2024.10446229

[25] Ziqian Ning, Yuepeng Jiang, Pengcheng Zhu, Jixun Yao, Shuai Wang, Lei Xie, and Mengxiao Bi. 2023. DualVC: Dual-mode Voice Conversion using Intra-model Knowledge Distillation and Hybrid Predictive Coding. In *Interspeech 2023*. 2063–2067. doi:10.21437/Interspeech.2023-1157

[26] William Peebles and Saining Xie. 2023. Scalable Diffusion Models with Transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 4195–4205.

[27] Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. 2018. FiLM: Visual Reasoning with a General Conditioning Layer. *Proceedings of the AAAI Conference on Artificial Intelligence* 32, 1 (Apr. 2018). doi:10.1609/aaai.v32i1.11671

[28] Kaizhi Qian, Yang Zhang, Shiyu Chang, Xuesong Yang, and Mark Hasegawa-Johnson. 2019. AutoVC: Zero-Shot Voice Style Transfer with Only Autoencoder Loss. In *Proceedings of the 36th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 97)*, Kamalika Chaudhuri and Ruslan Salakhutdinov (Eds.). PMLR, 5210–5219. https://proceedings.mlr.press/v97/qian19c.html

[29] Zengyi Qin, Wenliang Zhao, Xumin Yu, and Xin Sun. 2024. OpenVoice: Versatile Instant Voice Cloning. arXiv:2312.01479 [cs.SD] https://arxiv.org/abs/2312.01479

[30] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine Mcleavey, and Ilya Sutskever. 2023. Robust Speech Recognition via Large-Scale Weak Supervision. In *Proceedings of the 40th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 202)*, Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (Eds.). PMLR, 28492–28518. https://proceedings.mlr.press/v202/radford23a.html

[31] Chandan K A Reddy, Vishak Gopal, and Ross Cutler. 2021. Dnsmos: A Non-Intrusive Perceptual Objective Speech Quality Metric to Evaluate Noise Suppressors. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 6493–6497. doi:10.1109/ICASSP39728.2021.9414878

[32] Takaaki Saeki, Detai Xin, Wataru Nakata, Tomoki Koriyama, Shinnosuke Takamichi, and Hiroshi Saruwatari. 2022. UTMOS: UTokyo-SaruLab System for VoiceMOS Challenge 2022. In *Interspeech 2022*. 4521–4525. doi:10.21437/Interspeech.2022-439

[33] Berrak Sisman, Junichi Yamagishi, Simon King, and Haizhou Li. 2021. An Overview of Voice Conversion and Its Challenges: From Statistical Modeling to Deep Learning. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 29 (2021), 132–157. doi:10.1109/TASLP.2020.3038524

[34] Chengyi Wang, Sanyuan Chen, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, Lei He, Sheng Zhao, and Furu Wei. 2023. Neural Codec Language Models are Zero-Shot Text to Speech Synthesizers. arXiv:2301.02111 [cs.CL] https://arxiv.org/abs/2301.02111

[35] Disong Wang, Liqun Deng, Yu Ting Yeung, Xiao Chen, Xunying Liu, and Helen Meng. 2021. VQMIVC: Vector Quantization and Mutual Information-Based Unsupervised Speech Representation Disentanglement for One-Shot Voice Conversion. In *Interspeech 2021*. 1344–1348. doi:10.21437/Interspeech.2021-283

[36] Tianrui Wang, Meng Ge, Zhikang Niu, Cheng Gong, Chunyu Qiang, Haoyu Wang, Zikang Huang, Ziyang Ma, Xiaobao Wang, Xie Chen, Longbiao Wang, and Jianwu Dang. 2025. A Progressive Generation Framework with Speech Pre-trained Model for Expressive Voice Conversion. In *2025 IEEE International Conference on Multimedia and Expo (ICME)*. 1–6. doi:10.1109/ICME59968.2025.11209157

[37] Zhichao Wang, Yuanzhe Chen, Xinsheng Wang, Lei Xie, and Yuping Wang. 2024. StreamVoice: Streamable Context-Aware Language Modeling for Real-time Zero-Shot Voice Conversion. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Lun-Wei Ku, Andre Martins, and Vivek Srikumar (Eds.). Association for Computational Linguistics, Bangkok, Thailand, 7328–7338. doi:10.18653/v1/2024.acl-long.396

[38] Yang Yang, Yury Kartynnik, Yunpeng Li, Jiuqiang Tang, Xing Li, George Sung, and Matthias Grundmann. 2024. STREAMVC: Real-Time Low-Latency Voice Conversion. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 11016–11020. doi:10.1109/ICASSP48485.2024.10446863

[39] Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. 2022. SoundStream: An End-to-End Neural Audio Codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 30 (2022), 495–507. doi:10.1109/TASLP.2021.3129994

[40] Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J. Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu. 2019. LibriTTS: A Corpus Derived from LibriSpeech for Text-to-Speech. In *Interspeech 2019*. 1526–1530. doi:10.21437/Interspeech.2019-2441

[41] Hanlin Zhang, Daxin Tan, Dehua Tao, Xiao Chen, Haochen Tan, Yunhe Li, Yuchen Cao, Jianping Wang, and Linqi Song. 2026. DSA-Tokenizer: Disentangled Semantic-Acoustic Tokenization via Flow Matching-based Hierarchical Fusion. arXiv:2601.09239 [cs.SD] https://arxiv.org/abs/2601.09239

[42] Xueyao Zhang, Xiaohui Zhang, Kainan Peng, Zhenyu Tang, Vimal Manohar, Yingru Liu, Jeff Hwang, Dangna Li, Yuhao Wang, Julian Chan, Yuan Huang, Zhizheng Wu, and Mingbo Ma. 2025. Vevo: Controllable Zero-Shot Voice Imitation with Self-Supervised Disentanglement. In *The Thirteenth International Conference on Learning Representations*. https://openreview.net/forum?id=anQDiQZhDP

[43] Yu Zhang, Baotong Tian, and Zhiyao Duan. 2025. Conan: A Chunkwise Online Network for Zero-Shot Adaptive Voice Conversion. arXiv:2507.14534 [eess.AS] https://arxiv.org/abs/2507.14534