Zero-Shot Sing Voice Conversion: a case study in embedding-based and token-based system

Anonymous ACL submission

Abstract

This study compares the sing voice conversion (SVC) system using embeddings extracted from SSL models and the subsequent discretized token representations. In the embedding-based experiments, we found that the number of samples per singer in the training data affects the degree of timbre leakage in the SVC model. Many previously considered effective methods for preventing timbre leakage warrant further examination regarding their specific performance sources. Additionally, we proposed an SVC system utilizing token representations, clustering vast amounts of data into a large codebook to extract phoneme representations that separate timbre information. We preliminarily explored the impact of the SSL model layer depth on phoneme representations and investigated how the number of codebooks affects the richness of these representations. Furthermore, this research advances zero-shot SVC and sets the stage for future work on discrete speech representation, emphasizing the preservation of rhyme.

1 Introduction

Singing Voice Conversion (SVC), a specific type of voice conversion transforms a singing voice from one singer to another while retaining the original melody, lyrics, and musical nuances. The aim is to produce a new recording that convincingly simulates the target singer's performance without altering the musical composition. This paper explores a zero-shot any-to-any SVC framework noted for its robust generalization.

A classic SVC system decodes singing into three components: content, pitch, and singer. During training, the system attempts to reconstruct the singing using this information. In inference, by substituting the singer's information with that of the target singer, sing voice conversion is achieved (Li et al., 2023; Liu et al., 2021). However, if the content contains some singer information, the model may partially reconstruct the singing using only the content during training, leading to the issue of timbre leakage.

Some VC systems use standard Phonetic PosteriorGrams (PPG) as the content encoder (Li et al., 2021), effectively capturing content in standard speech and separate singer information. However, given the complex rhythms and techniques in singing, standard Phonetic PosteriorGrams (PPG) inadequately capture essential content. Linguistic research highlights the International Phonetic Alphabet (IPA)'s limitations in capturing transitions between vowels and consonants, the effects of articulator interactions, and the inconsistent representation of phonetic features such as voicing and nasalization. Furthermore, the IPA inadequately reflects the nuanced phonological effects of articulation, limiting its effectiveness in detailing the intricate relationships among speech sounds (Ladefoged and Halle, 1988; Browman and Goldstein, 1992; Goldstein and Fowler, 2003; Mücke et al., 2020).

Content features from self-supervised learning (SSL) models (Oord et al., 2018) like WavLM (Chen et al., 2022) and Hubert (Hsu et al., 2021) provide extensive information but also capture speaker-specific details. To ensure the reconstructed audio's timbre closely matches the target's, it's crucial to reduce timbral characteristics in content features. ContentVec (Qian et al., 2022) utilizes a teacher-student approach with Hubert to separate speaker information from content, effectively reducing timbre leakage. FreeVC (Li et al., 2023) employs a spectrogram resizing technique and a bottleneck module with WavLM-extracted features, achieving good timbre replication in in-sample tests but less so in out-of-sample scenarios. KNN-VC (Baas et al., 2023), using clustered features from WavLM for the target speaker, significantly mitigates timbre leakage with minimal target

speaker data within the RVC project[1], though its performance on zero-shot conversion is hindered by its dependence on the amount of target data.

We chose So-VITS-SVC[2] as the foundational framework and utilized the SSL model Hubert to extract content information for the SVC model, exploring the impact of different configurations on SVC performance at both the embedding-based and token-based levels.

In the embedding-based study, we found that differently constructed training datasets have varying effects on the model's timbre leakage issue. Specifically, training datasets with fewer recordings per singer are prone to timbre leakage, indicating that the strong in-domain voice conversion performance observed in prior research—using open-source datasets with limited speaker variety but many recordings per individual—likely benefits from the extensive data on target speakers. However, such studies fall short in achieving effective out-of-domain, zero-shot voice conversion, highlighting the need to address timbre leakage to create versatile SVC models trained on large, diverse datasets.

In the token-based SVC system study, we considered clustered features from multi-speaker data through the Hubert model as a representation approach independent of existing linguistic frameworks. The specific motivations are twofold: firstly, the Hubert model is a self-supervised learning (SSL) model optimized for extracting audio representations, highly beneficial for downstream tasks like automatic speech recognition, due to its ability to capture extensive content information. The impact of content loss through clustering is mitigated with an adequate number of cluster centers. Secondly, similar pronunciations across different speakers tend to converge within the same cluster centers, effectively abstracting pronunciation across speakers in a given frame while eliminating speaker-specific details.

Based on the two motivations mentioned above, we specifically explored the impact of SSL embeddings from different layers on the phoneme representation capabilities of clustered tokens. Additionally, we investigated how the number of cluster centers affects the richness of phoneme representations.

Utilizing the knowledge gained from the above exploration, we employed clustering-based phoneme representation without relying on textual data, incorporating fundamental frequency values and speaker embeddings at the utterance level for singing voice reconstruction. Ultimately, we trained a powerful SVC model on 10,000 hours of audio data from 500,000 singers, achieving strong out-of-domain capabilities and reaching the zero-shot any-to-any singing voice conversion objective. Further details and demonstrations are accessible on our demo page [3].

In summary, this paper makes three key contributions:

- We found that the training dataset structure influences the SVC system's performance, as sufficient samples from a single singer reduce timbre leakage, whereas a few samples per singer increase the risk of leakage despite larger overall training data, particularly in the embedding-based SVC system.
- We found that SSL embeddings from deeper layers in a token-based SVC system exhibit reduced acoustic representation capabilities due to the influence of the ASR task.
- We compared the performance of a token-based SVC system with varying numbers of clustered centers and explored the structure of the resulting clusters.

2 Proposed Approach

As depicted in Figure 1, our research adopts a methodology consistent with the core architecture of non-autoregressive audio generation models. This approach includes a content encoder, an acoustic model, and a Variational Autoencoder (VAE) (Mescheder et al., 2017) focused on audio reconstruction. Furthermore, we incorporate WeSpeaker (Wang et al., 2023) as a speaker encoder, significantly improving the model's ability to identify and replicate speaker-specific features.

The training phase, described by Equation (1), includes a two-step procedure: first, separating the original audio's components, and then, reconstructing the audio. In contrast, the voice conversion inference process, captured by Equation (2), focuses on producing audio in the target timbre. This involves extracting content information from

[1] https://github.com/RVC-Project/

[2] https://github.com/svc-develop-team/so-vits-svc

[3] This paper demo page: https://expdemos.github.io/zeroshotSVC/

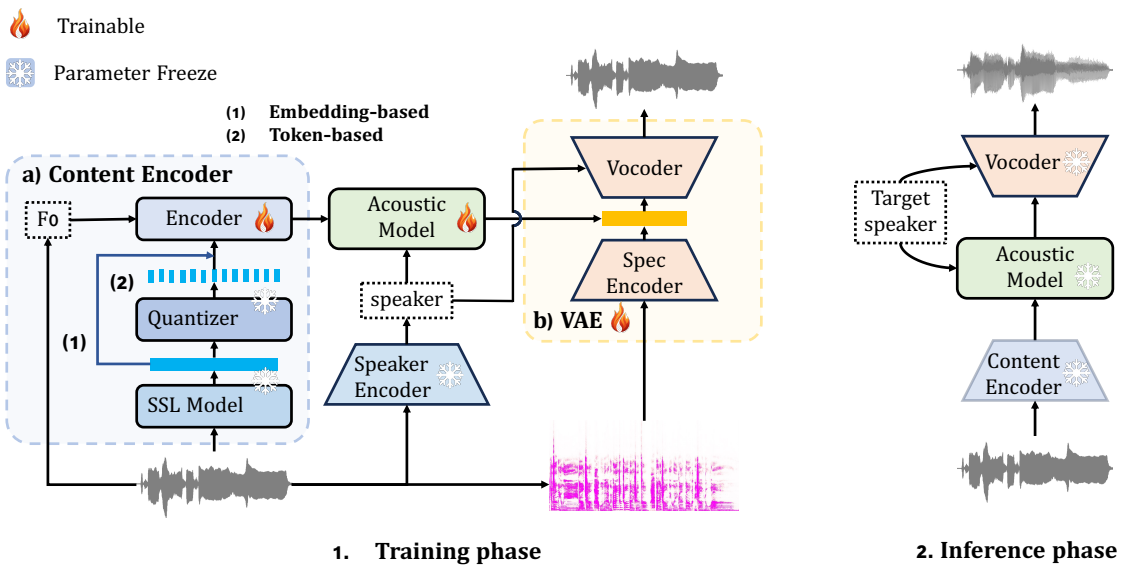

Figure 1: Basic Framework.

the source audio and applying the target singer's speaker embedding to this content.

$$Wav'_s = \mathbf{VC}(\mathbf{KM}^{\mathbf{o}}_{\mathbf{num}}(\mathbf{H}(Wav_s)), F0_s, S_s) \quad (1)$$

$$Wav'_t = \mathbf{VC}(\mathbf{KM}^{\mathbf{o}}_{\mathbf{num}}(\mathbf{H}(Wav_s)), F0_t, S_t) \quad (2)$$

Within this study, $\mathbf{H}$ represents the features extracted by the Hubert model, while $\mathbf{KM}^{\mathbf{o}}_{\mathbf{num}}$ denotes an optional k-means clustering algorithm with a specific number of cluster centers. The voice conversion (VC) module, $\mathbf{VC}$, illustrated in Figure 1, integrates a trainable setup comprising an encoder in the content encoder, an acoustic model, and a Variational Autoencoder (VAE). The symbols Wav_s and Wav_t are used for the source and target waveforms, respectively. The output waveform is marked as Wav'. Pitch data from the source waveform is shown as $F0_s$, with $F0_t$ indicating the adapted pitch for the target waveform, adjusting $F0_s$ to align with the fundamental frequency of Wav_t. Lastly, speaker embeddings for the source and target are labeled as S_s and S_t, respectively.

2.1 Content Encoder

The Content Encoder handles various content information, converting it into a format suited for the acoustic model. It has two content-representing scenarios, one utilizes the Hubert model to extract audio latent features, and another discretizes these latent features by a quantizer. Importantly, the self-supervised learning model, Hubert, along with the quantizer, are not involved in the training phase of the voice conversion module.

2.1.1 Hubert

Our study retrained the Hubert model on a vast, internally curated dataset featuring 300,000 hours of multi-domain Chinese audio. We utilized the Hubert Large (Hsu et al., 2021) architecture, which includes seven CNN layers for downsampling and 24 transformer layers. The Hubert model underwent fine-tuning on ASR tasks after pre-training. Since the Hubert model was fine-tuned on ASR tasks using CTC after pre-training, we hypothesize that the embeddings from the last layer, specifically layer 24, contain text-related information rather than phoneme-related content. Therefore, we selected the embeddings from layer 22 for the construction of the embedding-based system. Additionally, we validated the performance of layers 22 and 24 in terms of content representation accuracy within the token-based SVC system.

2.1.2 Quantizer

The function of the quantizer within the study is to discretize features extracted by the SSL model. We choose MiniBatchKMeans (Chavan et al., 2015) as our quantization method for two primary reasons. Firstly, we posit that applying K-means clustering to a substantial multi-speaker dataset could identify cluster centers that encapsulate generalized phoneme information, effectively omitting speaker-specific nuances. Such discrete representations could then serve effectively as phoneme rep-

resentations, especially since MiniBatchKMeans, when used with sufficiently large batch sizes, approximates the outcomes of traditional KMeans closely. Secondly, evidence from Base TTS (Łajszczak et al., 2024) indicates that vector quantization (VQ) (Gârbacea et al., 2019) techniques reliant on back-propagation are significantly influenced by the distribution of data, which complicates the clustering of less prevalent phonetic features. However, our findings suggest that MiniBatchKMeans, particularly when trained with maximally large batch sizes, can mitigate the effects of an imbalanced data distribution.

To accommodate the processing of exceptionally large batch sizes swiftly, we develop a GPU-accelerated version of MiniBatchKMeans. This innovation enabled us to efficiently cluster data into 10,000 cluster centers with an unprecedented batch size of 1.5 million.

2.1.3 Encoder

The Encoder in this study is tasked with encoding a blend of diverse content information, with our experiments specifically focusing on the reconstruction of pitch information alongside the primary content. We employ `RMVPE` (Wei et al., 2023) to extract pitch information $F0$, which is then converted to Log-F0 and fed into an embedding layer. And the the encoder directly adds the embedded pitch and content information together and then encoders them by multi-head attention.

2.2 Acoustic Model

The Acoustic Model is responsible for transforming content information into an intermediate representation for the VAE. We employ the Glow model G as our acoustic framework (Kim et al., 2020). The core design features a sequence of affine coupling layers, which are constructed using multiple residual blocks from WaveNet (Van Den Oord et al., 2016). In the training phase, the model translates the posterior distribution $q(z)$, conditioned on the singer s, to the prior distribution $p(\hat{z})$ through the inverse of G, with KL divergence serving as the loss function. During the inference stage, Glow G utilizes the prior distribution $p(\hat{z})$, conditioned on the singer s, to produce a predicted distribution $q(z^*)$. It's important to note that the choice of acoustic model is versatile; options are not confined to this structure alone but can extend to encompass other types, including flow and diffusion models (Ho et al., 2020), among others.

2.3 VAE

The VAE (Mescheder et al., 2017) works as a waveform reconstructor, compressing the audio waveform into a compact, low-dimensional representation before reconstructing it back into its original waveform form. This compact representation serves as the crucial output for the Acoustic Model. The process starts by transforming the singing from a waveform into a spectrogram. We adopt WaveNet (Van Den Oord et al., 2016) as the spectral encoder (the encoder of the VAE) and derive the mean and variance, which are then utilized to sample the posterior distribution $q(z)$ from a Gaussian distribution. This sampled distribution $q(z)$ is subsequently used for training the acoustic model. HiFiGAN, enhanced with a Neural Synthesis Filter (NSF), serves as the decoder to reconstruct the waveform from $q(z)$. The decoder is trained using a combination of reconstruction loss, adversarial loss, and feature mapping loss, alongside the acoustic model, wave encoder, and posterior encoder. It's worth noting that the choice of vocoder is not restricted to nsf-HiFiGan; a variety of vocoders, including various iterations of HiFiGan (Lee et al., 2022; Kong et al., 2020), can be effectively utilized.

2.4 Speaker Encoder

The speaker encoder is designed to capture speaker-specific characteristics from singing voices. We choose ResNet34[4] (He et al., 2016), sourced from the WeSpeaker open-source framework, to serve as the speaker encoder. This encoded speaker information plays a crucial role in the functioning of both the acoustic model and the vocoder, ensuring that the unique vocal traits of each singer are accurately preserved and reflected in the synthesized audio.

3 Experiments

3.1 Ablation Experiments

We conduct multiple ablation studies:

a. Embeddings only Sing Reconstruction

This experiment aims to reconstruct singing solely using SSL embeddings to validate that the embeddings themselves contain a substantial amount of timbre information. This experiment was also used to directly assess the phoneme representation capabilities of SSL embeddings from

[4]https://wespeaker-1256283475.cos.ap-shanghai.myqcloud.com/models/voxceleb/voxceleb_resnet34.zip

different layers. In practical implementation, both f0 and speaker embeddings are set as tensors filled with ones.

$$Wav'_s = \mathbf{VC}(\mathbf{H}(Wav_s)), \text{train} \quad (3)$$

$$Wav'_s = \mathbf{VC}(\mathbf{H}(Wav_s)), \text{infer} \quad (4)$$

b. Embeddings-based SVC

This experiment examines the impact of differently constructed training datasets on timbre leakage.

$$Wav'_s = \mathbf{VC}(\mathbf{H}(Wav_s), F0_s, S_s), \text{train} \quad (5)$$

$$Wav'_s = \mathbf{VC}(\mathbf{H}(Wav_s), F0_s, S_t), \text{infer1} \quad (6)$$

$$Wav'_t = \mathbf{VC}(\mathbf{H}(Wav_s), F0^t_s, S_t), \text{infer2} \quad (7)$$

c. Token-based SVC

This experiment explores the representation capabilities of tokens under different configurations through the performance of a token-based SVC system.

$$Wav'_s = \mathbf{VC}(\mathbf{KM_{num}}(\mathbf{H}(Wav_s)), F0_s, S_s), \text{train} \quad (8)$$

$$Wav'_s = \mathbf{VC}(\mathbf{KM_{num}}(\mathbf{H}(Wav_s)), F0_s, S_t), \text{infer1} \quad (9)$$

$$Wav'_t = \mathbf{VC}(\mathbf{KM_{num}}(\mathbf{H}(Wav_s)), F0^t_s, S_t), \text{infer2} \quad (10)$$

Experiments **b.** and **c.** feature two inference modes based on whether the pitch (F0) has been modified through conversion. We assess the timbre retention by comparing the speaker cosine similarity between the generated waveform without pitch conversion Wav' and the source waveform Wav_s denoted as $SrcSIM$. This evaluation helps to determine the extent of timbre leakage. Similarly, the effectiveness of voice conversion is evaluated by comparing the speaker cosine similarity between the pitch-converted generated waveform Wav' and the target waveform Wav_t, referred to as $TgtSIM$. Note that pitch conversion involves adding the difference between the mode of the F0 of Wav_t and Wav_s to $F0_s$, resulting in $F0^t_s$. The formula for calculating $F0_t$ is as follows:

$$F0^{\mathrm{t}}_{\mathrm{s}} = F0_{\mathrm{s}} \times \frac{mean(F0_{\mathrm{t}})}{mean(F0_{\mathrm{s}})} \quad (11)$$

3.2 Dataset

The data utilized for these ablation studies are segregated into two distinct datasets. Dataset A includes 50 hours of singing data acquired from an external source, featuring recordings from 100 singers, with each singer contributing an average of half an hour of samples. This dataset was purchased from a public source. Conversely, Dataset B is comprised of 200 hours of singing data, internally collected from 10,000 non-professional singers, with each contributing approximately one minute of samples.

For the testing phase, the source samples are drawn from an internal collection distinct from Datasets A and B. This test dataset consists of 30-second clips from 80 different songs performed by 80 unique singers. The target speakers chosen for evaluating the test samples are selected for their diversity, including two male and two female singers, each represented by 10-second clips of their singing.

The collection of internal data was conducted in accordance with user agreements, and user identity data has been anonymized.

3.3 Evluation Metrics

3.3.1 Objective Metrics

We use Singer Similarity (SSIM) to assess the effectiveness of voice conversion by extracting speaker embeddings with a speaker verification (SV) model and computing the cosine similarity.

As mentioned earlier, we need to assess the similarity between the converted song and both the target ($TgtSIM$) and source ($SrcSIM$) singing to determine the presence of timbre leakage. To further evaluate timbre leakage, we propose a new evaluation metric based on the two aforementioned similarities: the Target-Source Ratio TSR, as shown in Equation 12.

$$TSR = 100 \cdot \left(\frac{TgtSIM}{SrcSIM} - 1 \right) \quad (12)$$

3.3.2 Subjective Metrics

We use a 5-point (1-bad, 2-poor, 3-fair, 4-good, 5-excellent) Mean Opinion Score (MOS), which measures the naturalness. We selected 20 samples for these subjective evaluations and invited 25 evaluators to assess them.

It is noted that since each experiment is conducted up to the 100,000th training step, the vocoder has not been fully optimized, resulting in

the production of noisy singing voice outputs that could potentially influence MOS ratings during the ablation study phase. However, given that the training steps remained constant across all experiments, we consider the MOS scores to be comparable at this stage of the research. A higher TSR indicates that the similarity between the converted singing and the target is greater than the similarity with the source, suggesting a smaller degree of timbre leakage.

4 Discussions

4.1 Timbre Leakage

As indicated in Table 1, Experiment **a** reveals that the Hubert features successfully capture a wealth of audio detail, including speaker identity information. Therefore, to ensure the similarity of timbre in the converted voice, it is crucial to disentangle speaker information from content features within representations from the Hubert model.

By comparing the results of Experiment **b** using the embeddings from layer 22 for the SVC system on both Dataset A and Dataset B, it demonstrates that singing voice conversion models trained on datasets with limited recordings per singer exhibit more severe timbre leakage, even when trained on a broader dataset. This highlights the critical role of extensive per-speaker data in mitigating timbre leakage, particularly for models trained on publicly available datasets. Moreover, the findings indicate that the similarity to the target voice ($TgtSIM$) is less than the similarity to the source voice ($SrcSIM$) observed in Experiment **a**, stressing the importance of addressing timbre leakage to enable zero-shot SVC with novel, unseen data.

A comparison between Experiments **c** and **b** confirms that the token-based approach introduced in this study effectively reduces timbre leakage, and even outperformed the SVC system based on ContentVec which trained using complex supervised methods, representing a significant improvement in voice conversion tasks.

4.2 Hubert Layer

The results in Table 2 arise from an initial exploration of the most effective layer within the Hubert model for extracting features, focusing on the comparative efficacy of layers 22 and 24 as shown in Figure 2.

We conducted the experiments from two perspectives. The first is to validate the reconstruction capabilities of different Hubert layers, as demonstrated in Experiment **a**. The second is to assess the naturalness of content representation in the token-based SVC system across different layers, as shown in Experiment **c**.

Experiment **a.** demonstrates that, although the 22nd and 24th layers contain comparable speaker information, the 22nd layer's embeddings are more conducive to reconstructing prosody. This advantage is primarily due to the lack of explicit pitch integration in the reconstruction process, which relies heavily on pitch cues within the Hubert embeddings for recreating singing. On the other hand, embeddings from the 24th layer provide less detailed prosodic information, resulting in reconstructions with a more mechanical sound.

Additionally, experiment **c.** emphasizes the superiority of the 22nd layer in minimizing timbre leakage and improving speaker conversion accuracy. In contrast, reconstructions from the 24th layer are noted for their pronunciation of mechanical sense, as determined through subjective evaluation. Thus, this investigation firmly establishes the 22nd layer as the optimal choice for extracting content embeddings.

The possible reason for this performance gap is that the Hubert model was fine-tuned on ASR tasks using CTC loss. This results in embeddings closer to the final layer being more aligned with text rather than phonemes.

Table 1: Evaluations of timbre leakage. **AT** denotes ablation experiments type. H22 means embeddings extracted from the 22nd layer of Hubert. KM4096 menas Kmeans with 4096 cluster centers. SrcSIM denotes speaker cosine similarity between Wav'_s and Wav_s. TgtSIM denotes speaker cosine similarity between Wav'_t and Wav_t.

AT	Content	Data	SrcSIM↓	TgtSIM↑	TSR↑
a	H22	B	0.8149	\	\
b	H22	A	0.7350	0.7819	6.38
	H22	B	0.7766	0.7914	1.91
	ContentVec	B	0.7073	0.8216	16.75
c	H22+KM4096	B	**0.6791**	**0.8517**	**20.42**

Table 2: Evaluations of Hubert layer comparison experiments. Subjective MOS with 95% confidence intervals are shown.

AT	Content	Data	SrcSIM↓	TgtSIM↑	MOS↑
a	H22	B	0.8149	\	2.71 ± 0.09
	H24	B	0.8168	\	2.44 ± 0.12
c	H22+KM4096	B	**0.6791**	**0.8517**	3.83 ± 0.05
	H24+KM4096	B	0.6903	0.8475	3.50 ± 0.07

4.3 Clustering Centers

Table 3 presents an investigation into how varying the number of clustering centers affects the model's performance. The analysis shows that changes in the number of clusters have minimal impact on timbral fidelity, indicating the model's capability to consistently replicate timbre across different configurations. However, content fidelity demonstrates a clear relationship with the cluster count: more clusters lead to better reproduction of detailed rhythmic nuances. Decreasing the cluster count to 4096 resulted in a noticeable decline in capturing complex singing techniques and introduced minor issues with articulation clarity in fast speech sequences. Reducing the clusters further to 2048 intensified these articulation clarity issues, highlighting a significant reliance on the granularity of clustering for accurate articulation representation.

To further quantify the representation performance of tokens derived by different clustering models with different numbers of clustering centers, three metrics are adopted for analysis:

(1) AMD (Average Minimum Distance): This metric assesses how effectively the clustering has grouped similar features. It calculates the average Euclidean distance from a large set of Hubert features (1.5 million in this case) to their nearest cluster center. A smaller average distance indicates that the features are closer to their respective clusters, suggesting better clustering efficiency.

(2) MDC (Minimum Distance among Cluster Centers): This metric measures how distinctly different the clusters are from one another by calculating the smallest Euclidean distance between any two cluster centers. A larger minimum distance implies that the clusters are well-separated, which enhances the model's ability to distinguish between different phonemes.

(3) QDC (Quintile Distance among Cluster Centers): This metric evaluates how robust the cluster separation is in the presence of outliers. It calculates the fifth percentile of the minimum Euclidean distances between all cluster centers, meaning it looks at the distance that is greater than or equal to 5% of the smallest distances. This helps to understand how resilient the clusters are against extreme values that might distort the overall clustering quality.

The results from the MDC metrics suggest that an increased number of clustering centers leads to more specific categorization in certain detailed aspects. Meanwhile, the QDC indicates that the fundamental phoneme clustering does not undergo significant structural changes with the increase in the number of clustering centers. This implies that while additional clustering centers refine the granularity of classifications, the overarching phoneme cluster architecture remains stable, preserving the essential phonetic distinctions regardless of cluster quantity.

The AMD metric indicates that as the number of clusters increases, the average loss per frame due to clustering decreases. The MDC metric results indicate that increasing the number of clustering centers enhances specificity in categorizing detailed aspects. On the other hand, the QDC metric shows that the core structure of phoneme clustering remains largely unchanged with more clustering centers. This suggests that while more centers refine the detail of classifications, the fundamental structure of phoneme clusters stays consistent, ensuring that critical phonetic distinctions are maintained regardless of the number of clusters.

At the same time, as cluster number increases, the relative change in QDC metrics is more stable compared to MDC, which reveals stable phoneme cluster structuring despite increased cluster numbers.

To visually illustrate the minimum Euclidean distances between each center and other centers in the codebook obtained with different cluster center numbers, we plotted a quantile plot, as shown in the Figure 2. It can be observed that when the cluster center number is set to ten thousand, the overall centers are more distinctly separated, leading to better representation capabilities for less frequently occurring phonemes.

In conclusion, the timbral replication remains stable across configurations, which supports that our model achieves great robustness in timbre preservation. In contrast, content fidelity improves with increased cluster numbers, enhancing the detail in rhythmic nuances. Notably, reducing clusters to 4096 compromises complex singing techniques and clarity in rapid speech, with a further reduction to 2048 exacerbating articulation issues.

4.4 Final Model

For the final model, we compile a dataset from our internal archives, featuring 10,000 hours of singing from 500,000 individuals, with about a minute of data per person. We use a tenth of this dataset to train the clustering model with 10,000 centers via

Table 3: Evaluations of cluster centers.

(a) VC Evaluation with 22nd Hubert Layer Embeddings					
AT	**Content**	**Data**	**SrcSIM↓**	**TgtSIM↑**	**MOS↑**
c	KM10000	B	0.6923	0.8500	4.01 ± 0.04
	KM4096	B	**0.6791**	0.8517	3.83 ± 0.05
	KM2048	B	0.6811	**0.8535**	3.37 ± 0.07
(b) Cluster Evaluation					
AT	**Content**	**Data**	**AMD↓**	**MDC↑**	**QDC**
	KM10000	B	307.95	61592.32	43716.60
	KM4096	B	315.51	52782.81	39246.32
	KM2048	B	326.83	52428.30	39503.28

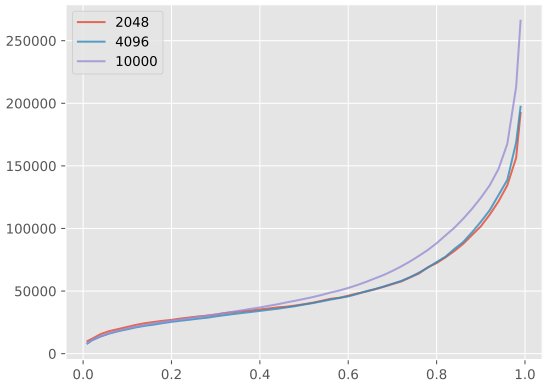

Figure 2: Quantile plot of the minimum Euclidean distances between each center and other centers in clustering models with varying cluster center numbers.

the MiniBatchKmeans method, with a batch size of 1.5 million. The training process is conducted over 10,000 iterations and takes about 48 hours on a single A100 GPU. The entire dataset is used to train the VC module. A Glow is adopted for the acoustic model, with nsf-HiFiGan as the vocoder. We modify the model parameters, setting gin_size to 256 and the vocoder's training slice length to 10,240, with a hop_length of 512. The final model has 30 million parameters. The VC module undergoes training on 40 A100 GPUs with 80 mini-batch sizes, requiring 350 hours for 800,000 steps.

Table 4 shows that our SVC model significantly distinguishes between source and target speakers in the test set, achieving high similarity with the target speaker and low similarity with the source. Content quality and reconstruction meet our standards. We also experiment with zero-filled tensors for speaker embedding, leading to a more neutral voice timbre as expected. However, this method results in reduced audio quality after conversion, due to the significant difference from the training sample embeddings.

Table 4: Final model performance. The empty speaker means inference with all zeros tensor as speaker embedding.

Speaker	**SrcSIM↓**	**TgtSIM↑**	**MOS↑**
Tgt	0.6658	0.8614	4.3 ± 0.05

5 Limitations

This work is a preliminary exploration of embedding-based and token-based SVC systems, conducted solely within the So-VITS-SVC framework. Additionally, our experiments revealed that the token-based SVC system may experience content reconstruction errors on non-training languages, specifically resulting in distortions of certain phonemes. This limitation restricts the applicability of the final model.

6 Conclusion

In this study, we explored both embedding-based and token-based zero-shot SVC systems within the So-VITS-SVC framework. Our embedding-based experiments revealed that the number of samples per singer in the training dataset significantly influences the extent of timbre leakage in the SVC model. This finding suggests that many previously effective methods for mitigating timbre leakage merit a deeper investigation into their specific performance characteristics.

Moreover, we introduced an SVC system that employs token representations by clustering extensive data into a large codebook, allowing for the extraction of phoneme representations that effectively separate timbre information. We also conducted preliminary analyses on how the depth of the SSL model layers affects phoneme representations and examined the impact of the number of codebooks on the richness of these representations.

This research not only advances the field of zero-shot SVC but also lays the groundwork for future studies on discrete speech representation, with a particular focus on maintaining rhyme.

References

Matthew Baas, Benjamin van Niekerk, and Herman Kamper. 2023. Voice conversion with just nearest neighbors. *arXiv preprint arXiv:2305.18975*.

Catherine P Browman and Louis Goldstein. 1992. Articulatory phonology: An overview. *Phonetica*, 49(3-4):155–180.

M Chavan, Asawari Patil, Lata Dalvi, and Ajinkya Patil. 2015. Mini batch k-means clustering on large dataset. *Int. J. Sci. Eng. Technol. Res*, 4(07):1356–1358.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, et al. 2022. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518.

Cristina Gârbacea, Aäron van den Oord, Yazhe Li, Felicia SC Lim, Alejandro Luebs, Oriol Vinyals, and Thomas C Walters. 2019. Low bit-rate speech coding with vq-vae and a wavenet decoder. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 735–739. IEEE.

Louis Goldstein and Carol A Fowler. 2003. Articulatory phonology: A phonology for public language use. *Phonetics and phonology in language comprehension and production: Differences and similarities*, pages 159–207.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460.

Jaehyeon Kim, Sungwon Kim, Jungil Kong, and Sungroh Yoon. 2020. Glow-tts: A generative flow for text-to-speech via monotonic alignment search. *Advances in Neural Information Processing Systems*, 33:8067–8077.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. 2020. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in neural information processing systems*, 33:17022–17033.

Peter Ladefoged and Morris Halle. 1988. Some major features of the international phonetic alphabet. *Language*, 64(3):577–582.

Mateusz Łajszczak, Guillermo Cámbara, Yang Li, Fatih Beyhan, Arent van Korlaar, Fan Yang, Arnaud Joly, Álvaro Martín-Cortinas, Ammar Abbas, Adam Michalski, et al. 2024. Base tts: Lessons from building a billion-parameter text-to-speech model on 100k hours of data. *arXiv preprint arXiv:2402.08093*.

Sang-gil Lee, Wei Ping, Boris Ginsburg, Bryan Catanzaro, and Sungroh Yoon. 2022. Bigvgan: A universal neural vocoder with large-scale training. *arXiv preprint arXiv:2206.04658*.

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)*, pages 1–5. IEEE.

Zhonghao Li, Benlai Tang, Xiang Yin, Yuan Wan, Ling Xu, Chen Shen, and Zejun Ma. 2021. Ppg-based singing voice conversion with adversarial representation learning. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7073–7077. IEEE.

Songxiang Liu, Yuewen Cao, Na Hu, Dan Su, and Helen Meng. 2021. Fastsvc: Fast cross-domain singing voice conversion with feature-wise linear modulation. In *2021 ieee international conference on multimedia and expo (icme)*, pages 1–6. IEEE.

Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. 2017. Adversarial variational bayes: Unifying variational autoencoders and generative adversarial networks. In *International conference on machine learning*, pages 2391–2400. PMLR.

Doris Mücke, Anne Hermes, and Sam Tilsen. 2020. Incongruencies between phonological theory and phonetic measurement. *Phonology*, 37(1):133–170.

A. Oord, Y. Li, and O. Vinyals. 2018. Representation learning with contrastive predictive coding. In *Proc. NeurIPS*.

Kaizhi Qian, Yang Zhang, Heting Gao, Junrui Ni, Cheng-I Lai, David Cox, Mark Hasegawa-Johnson, and Shiyu Chang. 2022. Contentvec: An improved self-supervised speech representation by disentangling speakers. In *International Conference on Machine Learning*, pages 18003–18017. PMLR.

Aaron Van Den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, Koray Kavukcuoglu, et al. 2016. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 12.

Hongji Wang, Chengdong Liang, Shuai Wang, Zhengyang Chen, Binbin Zhang, Xu Xiang, Yanlei Deng, and Yanmin Qian. 2023. Wespeaker: A research and production oriented speaker embedding

learning toolkit. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Haojie Wei, Xueke Cao, Tangpeng Dan, and Yueguo Chen. 2023. Rmvpe: A robust model for vocal pitch estimation in polyphonic music. *arXiv preprint arXiv:2306.15412*.