MeloCodec: Harnessing Melodic Priors for High-Fidelity Singing Voice Representation

Yizhong Geng*, Wenxin Fu*, Kecan Mao*, Qifei Li*, Yingming Gao*,
Ruimin Wang†, Chunfeng Wang†, Hao Li†, Ya Li*†, Wei Chen††
*Beijing University of Posts and Telecommunications, Beijing, China
†Li Auto, Beijing, China

***Abstract*—Neural audio codecs serve as fundamental tokenizers for LLM-based audio generation. While semantic priors are widely exploited to enhance linguistic intelligibility, the integration of explicit acoustic priors remains underexplored, limiting synthesis fidelity in frequency-sensitive domains. To address this gap, we introduce MeloCodec, a novel framework designed to effectively incorporate melodic priors—a critical form of acoustic information for singing. To address the optimization instability typically caused by the direct fusion of such explicit priors, we propose a "Tokenize-then-Fuse" paradigm that pre-trains a discrete melodic branch to lock in structures before feature fusion. To robustly realize this paradigm, we further propose a robust two-stage training strategy, which prevents codebook collapse and ensures stable convergence. Experiments show that MeloCodec outperforms baselines in singing voice representation, improving pitch consistency and enabling controllable pitch manipulation with minimal timbre degradation. Audio samples are available at https://anonymous.4open.science/api/repo/melocodec_demo-60EC/file/demo.html?v=42c197c7.**

***Index Terms*—Neural Audio Codec, Singing Voice Representation, Melodic Priors, Residual Vector Quantization**

I. Introduction

Neural audio codecs have fundamentally transformed audio representation learning. Early models focused on high-fidelity waveform reconstruction at extremely low bitrate as efficient alternatives to traditional DSP standards [1], [2]. Following the success of LLMs in NLP [3] and CV [4], neural codecs have shifted from pure compression to discrete token modeling: vector-quantized audio codes bridge continuous waveforms and autoregressive generation, turning codecs into foundational "tokenizers" for generative AI. Systems such as VALL-E [5], AudioLM [6], and MusicGen [7] rely on these discrete representations to capture complex acoustic dependencies, making the codec latent space quality and structure critical to generation performance.

To satisfy LLM-based generation demands, recent works actively integrate semantic priors into the quantization process. Typical systems like X-Codec [8], SpeechTokenizer [9], and BiCodec [10] fuse SSL vectors, while CosyVoice2 [11] leverages ASR supervision to enhance linguistic alignment. However, prioritizing abstract semantics often neglects the intricate acoustic details essential for frequency-sensitive domains like singing voice [12], [13]. Conversely, while approaches like FACodec [14] attempt to model acoustics via implicit latent disentanglement, they lack rigid structural guarantees, often relying on unstable multi-objective constraints that are prone to information leakage.

To overcome the limitations of semantic-only approaches and implicit disentanglement, we propose **MeloCodec**, a novel neural audio codec that explicitly integrates melodic priors into the quantization process. While recent emerging works have begun to explore explicit priors like Chromagrams for synthesis [15]–[17], simply concatenating them within a codec's latent space introduces a critical challenge: Shortcut Learning [18]. Since the acoustic branch (processing raw waveforms) inherently contains superset information of the melody branch, the decoder tends to ignore the explicit melodic features, rendering the fusion ineffective [19], [20]. Furthermore, the direct fusion of these deterministic priors with flexible neural latents causes severe optimization instability due to gradient conflicts [18], [19], [21]–[23].

To address these challenges, we introduce **MeloCodec**, a novel framework built upon a **"Tokenize-then-Fuse" paradigm**. By enforcing an Information Bottleneck, this paradigm tokenizes melodic priors into quantized tokens to lock in structures before feature fusion, thereby reducing timbral leakage and encouraging the acoustic stream to rely on the melodic branch. To robustly realize this, we further propose a two-stage training strategy that resolves feature conflicts, prevents codebook collapse, and ensures stable convergence where direct fusion fails. Ultimately, MeloCodec establishes a scalable blueprint for bridging interpretable physical models with generative neural networks, proving that imposing discrete bottlenecks is essential for structurally grounded LLM-based audio generation.

Our principal contributions are summarized as follows:

- We propose **MeloCodec**, a novel codec framework that integrates explicit melodic priors into the quantization loop. Unlike previous conditioning methods, we embed melodic structures directly into the discrete latent space, establishing a unified and disentangled token interface for LLM-based generation.
- We introduce a **"Tokenize-then-Fuse" paradigm** to incorporate explicit melodic priors via an Information Bottleneck, and devise a robust two-stage training strategy to address shortcut learning and optimization instability, ensuring stable convergence.

†Corresponding authors

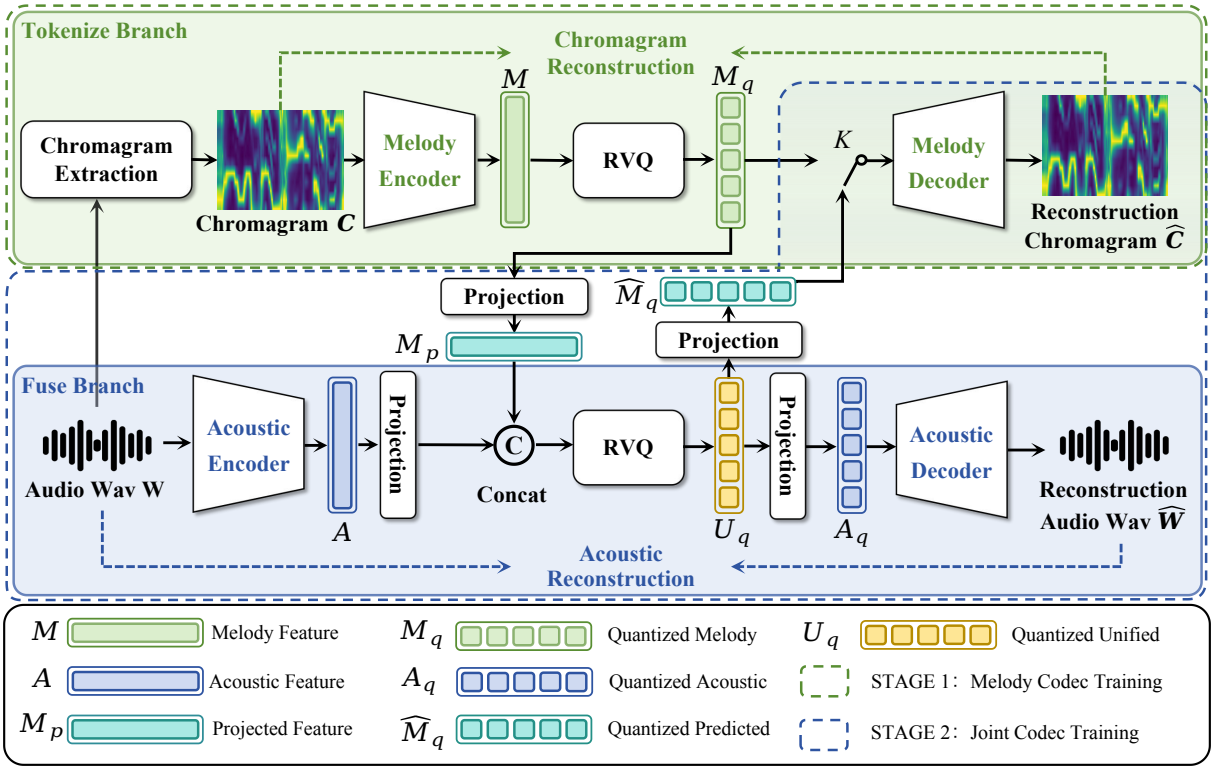

Fig. 1. The architecture of MeloCodec implementing the **"Tokenize-then-Fuse" paradigm**. The framework consists of a **Tokenize Branch** (Green) for discrete structural modeling and a **Fuse Branch** (Blue) for acoustic reconstruction. The switch K governs the data flow across the robust **two-stage training strategy**, where dashed boxes explicitly delimit the scope of active gradient updates: (1) In **Stage 1**, K routes the quantized melody M_q directly to the decoder to pre-train the bottleneck; (2) In **Stage 2**, K switches to the predicted auxiliary path ($\widehat{M}_q$), allowing gradients to update the Fuse Branch and fine-tune the Melody Decoder, while the pre-trained Melody Encoder and RVQ remains frozen.

- We demonstrate **State-of-the-Art Performance** through evaluations on singing voice datasets, showing that MeloCodec significantly outperforms baselines in pitch consistency. Furthermore, we verify that our model enables controllable pitch manipulation with minimal timbre degradation, confirming the effectiveness of our disentangled representation.

II. Method

A. Overview of Architecture

As shown in Fig. 1, MeloCodec decouples structural modeling from texture synthesis via a **"Tokenize-then-Fuse"** paradigm. Given waveform $\mathbf{x}$, the framework comprises two interacting streams: 1) The **Tokenize Branch** extracts melodic priors (chromagrams) and quantizes them into tokens $\mathbf{M}_q$ via RVQ, creating an information bottleneck to filter timbral noise; 2) The **Fuse Branch** encodes acoustic features $\mathbf{A}$ and integrates the pre-calculated $\mathbf{M}_q$ for high-fidelity reconstruction. To address the severe optimization instability of direct fusion, we employ a two-stage training strategy.

B. Tokenize Branch: Discrete Melodic Modeling

The **Tokenize Branch** leverages chromagrams to capture a speaker-invariant "melodic priors" [24]. By explicitly isolating structural progressions from spectral details, this branch creates a disentangled melodic reference for the system.

Given the input audio waveform $\mathbf{x} \in \mathbb{R}^T$, we first extract the chromagram $\mathbf{C} \in \mathbb{R}^{F \times N}$ (where $F = 12$ for pitch classes and N for time frames) using Short-Time Fourier Transform (STFT) followed by pitch-class projection: $\mathbf{C} = \Phi(\mathbf{x})$, where $\Phi(\cdot)$ acts as a physical filter to discard envelope information. The chromagram is then encoded by a convolutional melody encoder $\mathcal{E}_{mel}$ into continuous latent features $\mathbf{M}$:

$$\mathbf{M} = \mathcal{E}_{mel}(\mathbf{C}), \quad \mathbf{M} \in \mathbb{R}^{d \times N'} \tag{1}$$

where d is the hidden channel dimension and N' is the downsampled temporal resolution.

To discretize $\mathbf{M}$ into compact tokens, we apply RVQ with N_q levels and codebook $\mathcal{Z}$, accumulating residuals for a coarse-to-fine representation:

$$\mathbf{M}_q = \sum_{k=1}^{N_q} \mathbf{z}_k, \quad \text{where } \mathbf{z}_k = \text{Quantize}_k\left(\mathbf{M} - \sum_{j=1}^{k-1} \mathbf{z}_j\right) \tag{2}$$

This information bottleneck filters redundant noise, enforcing the latent space to capture essential structural semantics. A symmetric melody decoder $\mathcal{D}_{mel}$ reconstructs $\widehat{\mathbf{C}} =$

$\mathcal{D}_{mel}(\mathbf{M}_q)$. We pre-train this branch independently using the VQ-VAE objective:

$$\mathcal{L}_{stage1} = \|\mathbf{C} - \widehat{\mathbf{C}}\|_2^2 + \|\text{sg}[\mathcal{E}_{mel}(\mathbf{C})] - \mathbf{M}_q\|_2^2 \quad (3)$$

where sg[·] denotes the stop-gradient operator. This pre-training explicitly locks in structures, establishing a robust melodic "skeleton" that prevents codebook collapse in later joint optimization.

C. Fuse Branch: Heterogeneous Feature Fusion

The **Fuse Branch** integrates discrete priors ($\mathbf{M}_q$) with acoustic features. Crucially, fusing *quantized* tokens imposes an information bottleneck, preventing the shortcut learning and optimization instability inherent in continuous feature fusion.

The raw waveform $\mathbf{x}$ is processed by a convolutional acoustic encoder $\mathcal{E}_{ac}$ to yield dense features $\mathbf{A}$. Concurrently, the quantized melody tokens $\mathbf{M}_q$ from the Tokenize Branch are aligned to the acoustic space via a linear projection $\psi(\cdot)$:

$$\mathbf{A} = \mathcal{E}_{ac}(\mathbf{x}), \quad \mathbf{M}_p = \psi(\mathbf{M}_q) \quad (4)$$

where $\mathbf{A}, \mathbf{M}_p \in \mathbb{R}^{d \times N'}$, with d as the hidden channel dimension and N' as the downsampled temporal resolution matching the melodic stream. Fusing $\mathbf{M}_q$ rather than continuous $\mathbf{M}$ imposes a discrete information bottleneck, preventing shortcut learning where the decoder bypasses structural priors by relying on residual acoustic details.

The fused representation $\mathbf{U}$ is obtained via concatenation and then discretized into unified tokens $\mathbf{U}_q$ via a RVQ:

$$\mathbf{U} = \text{Concat}(\mathbf{A}, \mathbf{M}_p) \quad (5)$$

To enforce melodic integrity in $\mathbf{U}_q$, an auxiliary projection head $\phi(\cdot)$ predicts $\widehat{\mathbf{M}}_q = \phi(\mathbf{U}_q)$, which reconstructs the chromagram through the melody decoder. The auxiliary supervision loss is defined as:

$$\mathcal{L}_{aux} = \|\mathbf{C} - \mathcal{D}_{mel}(\widehat{\mathbf{M}}_q)\|_2^2 \quad (6)$$

This supervision backpropagates gradients from the pre-trained decoder, embedding acoustic information into the unified representation without dominating acoustic textures.

D. Training Strategy: Robust Two-Stage Optimization for "Tokenize-then-Fuse"

To robustly realize the **"Tokenize-then-Fuse" paradigm** and resolve the severe optimization instability arising from direct fusion between deterministic priors and flexible neural latents, we employ a robust two-stage training strategy. This strategy first isolates the Tokenize Branch to lock in structures for stability, and then jointly optimizes the Fuse Branch with anchored discrete priors, ensuring stable convergence where direct fusion fails.

In **Stage 1 (Melody Codec Pre-training)**, only the Tokenize Branch is active: the melody encoder $\mathcal{E}_{mel}$, RVQ, and decoder $\mathcal{D}_{mel}$ are fully trained using $\mathcal{L}_{stage1}$ (Eq. 3) to build a compact codebook of pitch transitions. This phase effectively locks in structures, keeping them free from acoustic interference.

In **Stage 2 (Joint Codec Training)**, we employ a partial freezing strategy: while $\mathcal{E}_{mel}$ and its RVQ remain frozen to anchor discrete priors, the melody decoder $\mathcal{D}_{mel}$ is fine-tuned. Empirically, we observe that a frozen decoder imposes overly rigid manifold constraints, causing optimization conflicts. Unfreezing $\mathcal{D}_{mel}$ allows it to accommodate distribution shifts in the unified tokens. This strategy effectively prevents codebook collapse and guarantees the stable convergence of the joint optimization. The Fuse Branch (Acoustic Stream) is optimized end-to-end. Specifically, $\mathcal{L}_{rec}$ comprises L1 waveform loss and multi-scale STFT loss. We employ a MSD for $\mathcal{L}_{adv}$, trained via a hinge loss objective with feature matching regularization. The total objective is:

$$\mathcal{L}_{stage2} = \mathcal{L}_{rec}(\mathbf{x}, \widehat{\mathbf{x}}) + \lambda_{adv}\mathcal{L}_{adv} + \lambda_{aux}\mathcal{L}_{aux} \quad (7)$$

This setup ensures robust supervision flow with minimal timbre degradation.

III. Experiments

A. Experimental Setup

Datasets and Baselines. Models were trained on an internal corpus of 5,500 hours of Mandarin singing voice (44.1kHz) and evaluated on the held-out Opencpop dataset [27]. To isolate architectural contributions, we compared MeloCodec against EnCodec [2], DAC [25], and X-Codec [8], all fine-tuned on our internal data. We also include MuCodec [26] (official weights) as a zero-shot reference for low-bitrate music, as its training code is unavailable.

Implementation Details. We use a codebook size of 1024. Performance is evaluated under **high bitrate** ($N_q = 8, \approx$6.0 kbps) and **low bitrate** ($N_q = 2, \approx$1.5 kbps) settings. Crucially, melodic tokens are priors only; reported bitrates are calculated exclusively on transmitted unified tokens $\mathbf{U}_q$. Training employed AdamW and standard EMA codebook updates on NVIDIA H800 GPUs. Inference uses a sliding window (50% overlap) to ensure continuity.

Metrics. Audio quality is measured via ViSQOL [28] and STOI [29]; pitch accuracy via F0-RMSE and Chroma-SIM; and timbre via SPK-SIM (ResNet-34) [30]. Subjective quality was assessed via a MUSHRA test [31] (15 experts, 30 samples). Controllability is quantified by Target F0-Corr under pitch shifting ($\pm 2, \pm 4$ semitones).

B. Reconstruction Performance

Objective Evaluation. Table I shows that in the high bitrate setting, MeloCodec outperforms the strong baseline DAC [25], confirming that integrating melodic priors enhances fidelity without interference.

The advantages are most typical in the low bitrate setting. As bandwidth tightens (N_q reduced to 2), traditional codecs degrade sharply in pitch consistency (e.g., Encodec's F0-RMSE nearly triples), suggesting standard RVQ neglects essential acoustic structural information. In contrast, MeloCodec remains robust (F0-RMSE 1.64). Furthermore, unlike MuCodec [26], which preserves melody but suffers in general quality (ViSQOL 2.62) due to low frame rates, MeloCodec

TABLE I

OBJECTIVE PERFORMANCE COMPARISON WITH STATE-OF-THE-ART BASELINES ON THE OPENCPOP TEST SET. TO ENSURE FAIR COMPARISON, WE EVALUATE MODELS UNDER TWO DISTINCT BITRATE SETTINGS: **HIGH BITRATE (≈ 6.0 KBPS)** AND **LOW BITRATE (≈ 1.5 KBPS)**. THE RESULTS HIGHLIGHT MELOCODEC'S ROBUSTNESS IN PRESERVING MELODIC STRUCTURE EVEN UNDER STRICTLY CONSTRAINED BANDWIDTHS.

Model	**Bitrate**	**Architecture**			**General Audio Quality**		**Melody & Pitch**		**Timbre**
	(kbps)	Codebook Size	N_q	Token Rate (Hz)	ViSQOL ↑	STOI ↑	F0-RMSE ↓	Chroma-Sim ↑	SPK-SIM ↑
high bitrate Setting (≈6.0 kbps)									
Encodec [2]	6.00	1024	8	75	3.92	0.82	1.92	0.93	0.97
DAC [25]	6.00	1024	8	75	3.97	0.83	1.23	0.96	0.98
X-Codec [8]	6.00	1024	8	75	3.34	0.81	1.35	0.95	0.96
MeloCodec (Ours)	6.00	1024	8	75	**4.08**	**0.85**	**0.96**	**0.97**	**0.99**
low bitrate Setting (≈1.5 kbps)									
Encodec [2]	1.50	1024	2	75	2.81	0.68	4.87	0.81	0.88
DAC [25]	1.50	1024	2	75	3.05	0.71	4.12	0.84	0.91
X-Codec [8]	1.50	1024	2	75	2.74	0.66	4.68	0.83	0.87
MuCodec [26]	1.33	10000	4	25	2.62	0.57	2.68	0.92	0.80
MeloCodec (Ours)	1.50	1024	2	75	**3.38**	**0.78**	**1.64**	**0.95**	**0.94**

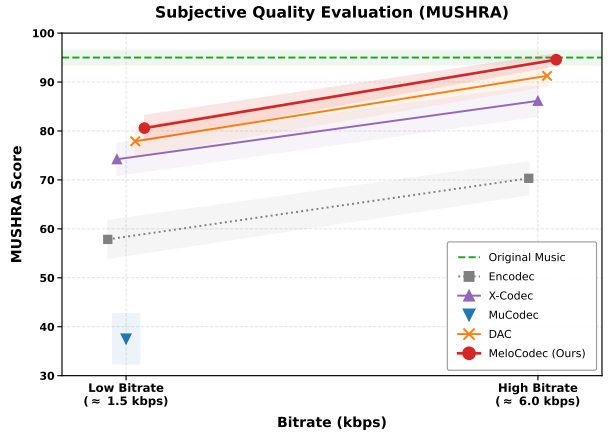

Fig. 2. Subjective MUSHRA evaluation results comparing MeloCodec with baselines across varying bitrates. The shaded areas represent the 95% confidence intervals.

effectively bridges this gap, delivering superior melodic integrity with minimal timbre degradation.

Subjective Evaluation. To corroborate the objective metrics, we conducted a MUSHRA (Multi-Stimulus Test with Hidden Reference and Anchor) listening test to evaluate perceptual quality. As illustrated in Figure 2, the subjective results align consistently with the objective data. At high bitrate, MeloCodec achieves the highest mean score, closely approaching the reference recording. Crucially, in the low-bitrate scenario, MeloCodec maintains robust performance superiority. While listeners reported noticeable artifacts and pitch instability in samples generated by baselines such as X-Codec and Encodec, MeloCodec retained clear melodic progressions and natural timbre. This underscores the effectiveness of our auxiliary melody supervision, which acts as a stable melodic prior to prevent codebook collapse under extreme compression rates.

C. Melodic Integrity and Pitch Analysis

To validate the model's capability in preserving and manipulating musical structures, we conduct a focused analysis on both reconstruction stability and latent disentanglement.

Reconstruction Stability. As detailed in Table I, MeloCodec consistently surpasses baselines in pitch accuracy. The advantage is most critical in the low bitrate setting (≈1.5 kbps). MeloCodec remains robust (F0-RMSE 1.64), acting as a structural anchor. This is visually corroborated in Figure 3, where our model captures subtle vibratos without the frequency jitter and octave errors observed in baseline reconstructions.

Disentanglement & Controllability. We evaluate controllability by shifting melodic priors ($\pm 2/4$ semitones) while keeping acoustic features fixed. As shown in Table III, **MeloCodec** maintains robust contour correlation (0.70–0.72), confirming effective melodic drive. However, the elevated RMSE (3.85–4.50) reveals a trade-off: while the contour shifts, absolute pitch precision remains partially constrained by the fixed acoustic branch. In contrast, Direct Fusion fails completely (RMSE 6.80), confirming its vulnerability to shortcut learning.

D. Ablation Study: Efficacy of Training Strategies

To validate the advantage of our proposed "Tokenize-then-Fuse" paradigm, we conducted ablation studies comparing MeloCodec against three variants: (1) **Acoustic-only**, a standard single-stream codec similar to DAC; (2) **+ Auxiliary Loss**, single-stream codec trained with additional melody reconstruction supervision; and (3) **Direct Fusion**, a dual-stream architecture trained end-to-end without the two-stage training strategy. The results are summarized in Table II.

Benefit of Melodic Supervision. Comparing the first two rows, adding an auxiliary melody loss yields a noticeable improvement in pitch accuracy, reducing F0-RMSE from 4.12 to 2.10 in the low-bitrate setting. This confirms that explicit supervisory signals can guide the encoder to retain more

TABLE II

ABLATION STUDY OF MELOCODEC TRAINING STRATEGIES ON THE OPENCPOP TEST SET. WE COMPARE DIFFERENT FUSION AND SUPERVISION METHODS USING CHROMAGRAM FEATURES. THE RESULTS HIGHLIGHT THAT THE PROPOSED "TOKENIZE-THEN-FUSE" PARADIGM (OURS) IS ESSENTIAL FOR STABILITY AND PERFORMANCE, ESPECIALLY AT LOW BITRATE.

Method	**Bitrate**	**Architecture**			**General Audio Quality**		**Melody & Pitch**		**Timbre**
	(kbps)	Codebook Size	N_q	Token Rate (Hz)	ViSQOL ↑	STOI ↑	F0-RMSE ↓	Chroma-Sim ↑	SPK-SIM ↑
high bitrate Setting (≈6.0 kbps)									
Acoustic-only	6.00	1024	8	75	3.97	0.83	1.23	0.96	0.98
+ Auxiliary Loss	6.00	1024	8	75	3.98	0.84	1.15	0.96	0.98
Direct Fusion	6.00	1024	8	75	3.52	0.79	2.30	0.93	0.96
Ours (Tokenize-then-Fuse)	6.00	1024	8	75	**4.08**	**0.85**	**0.96**	**0.97**	**0.99**
low bitrate Setting (≈1.5 kbps)									
Acoustic-only	1.50	1024	2	75	3.05	0.71	4.12	0.84	0.91
+ Auxiliary Loss	1.50	1024	2	75	3.25	0.75	2.10	0.91	0.93
Direct Fusion	1.50	1024	2	75	2.34	0.58	5.95	0.81	0.85
Ours (Tokenize-then-Fuse)	1.50	1024	2	75	**3.38**	**0.78**	**1.64**	**0.95**	**0.94**

TABLE III

EVALUATION OF DISENTANGLEMENT VIA PITCH SHIFTING. NOTE THAT WHILE ABSOLUTE PITCH PRECISION (RMSE) IS LIMITED BY ACOUSTIC CONSTRAINTS, MELOCODEC MAINTAINS SUPERIOR MELODIC CONTOUR (CORR) COMPARED TO DIRECT FUSION.

Method	**Shift**	**Pitch Control**		**Timbre Stability**	
	(Semi.)	Target F0-Corr ↑	Target F0-RMSE ↓	SPK-SIM ↑	Degrade % ↓
Direct Fusion	±2	0.40	4.10	0.88	7.6%
	±4	0.27	6.80	0.82	14.0%
MeloCodec	±2	**0.72**	**3.85**	**0.93**	**1.6%**
	±4	**0.70**	**4.50**	**0.91**	**3.9%**

pitch information. However, the gain in general audio quality (ViSQOL) is marginal, suggesting that soft constraints via loss functions alone are insufficient to fundamentally restructure the latent space.

Instability of Direct Fusion. We analyze the Direct Fusion strategy, representing the standard "One-stage" paradigm. To ensure fairness, this baseline employs learnable projection layers (MLP) and normalization to align heterogeneous modalities, avoiding simple concatenation. Despite these safeguards, the method exhibits catastrophic degradation (F0-RMSE worsens to 5.95), confirming severe optimization instability.

Diagnostic analysis of training dynamics reveals the root cause as modality collapse driven by feature competition. We observed two failure modes: 1) **Codebook Collapse**: Codebook utilization dropped to $< 5\%$ early in training, indicating the discrete bottleneck was bypassed. 2) **Optimization Divergence**: Crucially, the auxiliary chroma validation loss diverged continuously while training loss decreased. This confirms shortcut learning: dense acoustic gradients dominate optimization, causing the decoder to overfit acoustic textures rather than attending to rigid melodic priors. This verifies that standard architectural tuning (e.g., scaling or gating) is insufficient; rather, the proposed "Tokenize-then-Fuse" paradigm is structurally necessary to enforce effective melodic encoding.

Effectiveness of "Tokenize-then-Fuse". In contrast, our strategy achieves the best overall performance. By decoupling

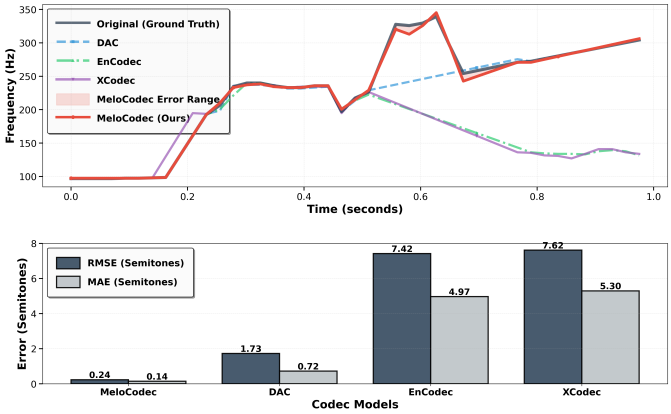

Fig. 3. Visual comparison of reconstructed pitch contours (F0). The upper panel shows F0 trajectories with the shaded area indicating MeloCodec's deviation. The lower panel presents error metrics RMSE and MAE.

training into two stages and fusing quantized melody tokens (M_q), we effectively resolve severe optimization instability. The pre-trained Melodic Stream explicitly locks in structures, allowing the Acoustic Stream to focus on texture synthesis. This is evident at low bitrate, where our method ensures stable convergence (F0-RMSE 1.64) compared to the collapse in Direct Fusion.

IV. CONCLUSION

In this paper, we presented **MeloCodec**, a dual-stream neural audio codec that explicitly integrates melodic priors to achieve high-fidelity singing voice representation. By addressing the severe optimization instability inherent in the direct fusion of features, our proposed "Tokenize-then-Fuse" paradigm effectively locks in structures via a pre-trained discrete branch, ensuring stable convergence and preventing codebook collapse. Results demonstrate that this strategy significantly outperforms baselines in pitch consistency with minimal timbre degradation. Crucially, our framework establishes a scalable blueprint for bridging interpretable physical

models with generative neural networks, proving that imposing discrete bottlenecks on deterministic priors is essential for robust feature integration. This opens new avenues for future research to extend the disentangled interface by incorporating additional explicit acoustic priors, such as rhythmic patterns and dynamic contours, paving the way for fully controllable and structurally grounded LLM-based audio generation.

V. Acknowledgment

The work was supported by the National Natural Science Foundation of China (NSFC) (No. 62271083), and the Key Project of the National Language Commission (No. ZDI145-81).

References

[1] Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi, "Soundstream: An end-to-end neural audio codec," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 495–507, 2021.

[2] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi, "High fidelity neural audio compression," *arXiv preprint arXiv:2210.13438*, 2022.

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al., "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[4] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever, "Zero-shot text-to-image generation," in *International conference on machine learning*. Pmlr, 2021, pp. 8821–8831.

[5] Chengyi Wang, Sanyuan Chen, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, et al., "Neural codec language models are zero-shot text to speech synthesizers," *arXiv preprint arXiv:2301.02111*, 2023.

[6] Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, et al., "Audiolm: a language modeling approach to audio generation," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 31, pp. 2523–2533, 2023.

[7] Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez, "Simple and controllable music generation," *Advances in Neural Information Processing Systems*, vol. 36, pp. 47704–47720, 2023.

[8] Zhen Ye, Peiwen Sun, Jiahe Lei, Hongzhan Lin, Xu Tan, Zheqi Dai, Qiuqiang Kong, Jianyi Chen, Jiahao Pan, Qifeng Liu, et al., "Codec does matter: Exploring the semantic shortcoming of codec for audio language model," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025, vol. 39, pp. 25697–25705.

[9] Xin Zhang, Dong Zhang, Shimin Li, Yaqian Zhou, and Xipeng Qiu, "Speechtokenizer: Unified speech tokenizer for speech large language models," *arXiv preprint arXiv:2308.16692*, 2023.

[10] Xinsheng Wang, Mingqi Jiang, Ziyang Ma, Ziyu Zhang, Songxiang Liu, Linqin Li, Zheng Liang, Qixi Zheng, Rui Wang, Xiaoqin Feng, et al., "Spark-tts: An efficient llm-based text-to-speech model with single-stream decoupled speech tokens," *arXiv preprint arXiv:2503.01710*, 2025.

[11] Zhihao Du, Yuxuan Wang, Qian Chen, Xian Shi, Xiang Lv, Tianyu Zhao, Zhifu Gao, Yexin Yang, Changfeng Gao, Hui Wang, et al., "Cosyvoice 2: Scalable streaming speech synthesis with large language models," *arXiv preprint arXiv:2412.10117*, 2024.

[12] Yichen Han, Xiaoyang Hao, Keming Chen, Weibo Xiong, Jun He, Ruonan Zhang, Junjie Cao, Yue Liu, Bowen Li, Dongrui Zhang, et al., "Quantize more, lose less: Autoregressive generation from residually quantized speech representations," *arXiv preprint arXiv:2507.12197*, 2025.

[13] Yizhong Geng, Jizhuo Xu, Zeyu Liang, Jinghan Yang, Xiaoyi Shi, and Xiaoyu Shen, "Scaling under-resourced tts: A data-optimized framework with advanced acoustic modeling for thai," in *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, 2025, pp. 593–604.

[14] Zeqian Ju, Yuancheng Wang, Kai Shen, Xu Tan, Detai Xin, Dongchao Yang, Yanqing Liu, Yichong Leng, Kaitao Song, Siliang Tang, et al., "Naturalspeech 3: Zero-shot speech synthesis with factorized codec and diffusion models," *arXiv preprint arXiv:2403.03100*, 2024.

[15] Xueyao Zhang, Junan Zhang, Yuancheng Wang, Chaoren Wang, Yuanzhe Chen, Dongya Jia, Zhuo Chen, and Zhizheng Wu, "Vevo2: Bridging controllable speech and singing voice generation via unified prosody learning," *arXiv preprint arXiv:2508.16332*, 2025.

[16] Ruiqi Li, Zhiqing Hong, Yongqi Wang, Lichao Zhang, Rongjie Huang, Siqi Zheng, and Zhou Zhao, "Accompanied singing voice synthesis with fully text-controlled melody," *arXiv preprint arXiv:2407.02049*, 2024.

[17] Huaize Liu, Wenzhang Sun, Donglin Di, Shibo Sun, Jiahui Yang, Changqing Zou, and Hujun Bao, "Moee: Mixture of emotion experts for audio-driven portrait animation," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 26222–26231.

[18] Junchuan Zhao, Wei Zeng, Tianle Lyu, and Ye Wang, "Comelsinger: Discrete token-based zero-shot singing synthesis with structured melody control and guidance," *arXiv preprint arXiv:2509.19883*, 2025.

[19] Adhiraj Banerjee and Vipul Arora, "Neural audio codecs for prompt-driven universal source separation," *arXiv e-prints*, pp. arXiv–2509, 2025.

[20] Frederik Bous and Axel Roebel, "Vasab: The variable size adaptive information bottleneck for disentanglement on speech and singing voice," *arXiv preprint arXiv:2310.03444*, 2023.

[21] Haici Yang, Inseon Jang, and Minje Kim, "Generative de-quantization for neural speech codec via latent diffusion," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 1251–1255.

[22] Bingsong Bai, Yizhong Geng, Fengping Wang, Cong Wang, Puyuan Guo, Yingming Gao, and Ya Li, "Hq-svc: Towards high-quality zero-shot singing voice conversion in low-resource scenarios," *arXiv preprint arXiv:2511.08496*, 2025.

[23] Wenzhang Sun, Zhenyu Wang, Zhangchi Hu, Chunfeng Wang, Hao Li, and Wei Chen, "Muse: A multi-agent framework for unconstrained story envisioning via closed-loop cognitive orchestration," *arXiv preprint arXiv:2602.03028*, 2026.

[24] Junan Zhang, Yunjia Zhang, Xueyao Zhang, and Zhizheng Wu, "Anyaccomp: Generalizable accompaniment generation via quantized melodic bottleneck," *arXiv preprint arXiv:2509.14052*, 2025.

[25] Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar, "High-fidelity audio compression with improved rvqgan," *Advances in Neural Information Processing Systems*, vol. 36, pp. 27980–27993, 2023.

[26] Yaoxun Xu, Hangting Chen, Jianwei Yu, Wei Tan, Rongzhi Gu, Shun Lei, Zhiwei Lin, and Zhiyong Wu, "Mucodec: Ultra low-bitrate music codec," *arXiv preprint arXiv:2409.13216*, 2024.

[27] Yu Wang, Xinsheng Wang, Pengcheng Zhu, Jie Wu, Hanzhao Li, Heyang Xue, Yongmao Zhang, Lei Xie, and Mengxiao Bi, "Opencpop: A high-quality open source chinese popular song corpus for singing voice synthesis," *arXiv preprint arXiv:2201.07429*, 2022.

[28] Andrew Hines, Jan Skoglund, Anil C Kokaram, and Naomi Harte, "Visqol: an objective speech quality model," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2015, no. 1, pp. 13, 2015.

[29] Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Transactions on audio, speech, and language processing*, vol. 19, no. 7, pp. 2125–2136, 2011.

[30] Brett Koonce, "Resnet 34," in *Convolutional neural networks with swift for tensorflow: image recognition and dataset categorization*, pp. 51–61. Springer, 2021.

[31] Michael Schoeffler, Fabian-Robert Stöter, Bernd Edler, and Jürgen Herre, "Towards the next generation of web-based experiments: A case study assessing basic audio quality following the itu-r recommendation bs. 1534 (mushra)," in *1st Web Audio Conference*, 2015, pp. 1–6.