VOCODEC: AN EFFICIENT LIGHTWEIGHT LOW-BITRATES SPEECH CODEC

Leyan Yang[1,2,†], *Ronghui Hu*[1,2,†], *Yang Xu*[1,2,†], *Jing Lu*[1,2,*]

[1]Key Laboratory of Modern Acoustics, Nanjing University, Nanjing 210093, Jiangsu, China
[2]NJU-Horizon Intelligent Audio Lab, Horizon Robotics, Beijing 100094, China

ABSTRACT

Recent advancements in end-to-end neural speech codecs enable compressing audio at extremely low bitrates while maintaining high-fidelity reconstruction. However, low computational complexity and low latency remain crucial for real-time communication. In this paper, we propose VoCodec, an audio codec model featuring a computational complexity of only 349.29 M multiply-accumulate operations per second (MAC/s) and a latency of 30 ms. Additionally, we cascade a neural network for speech enhancement at the front end to extend its capabilities of noise reduction and dereverberation. Experimental results demonstrate that the two systems deliver superior performance across multiple evaluation metrics.

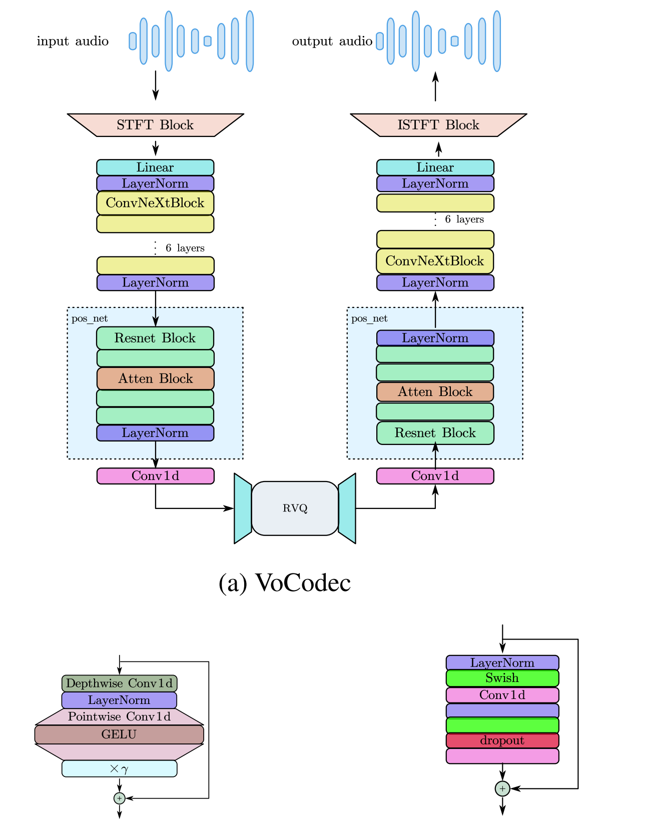
***Index Terms*—** audio codec, low computational resource, low bitrate, generative adversarial network, speech enhancement

1. INTRODUCTION

Recently, audio codec models have achieved significant progress in recovering high-quality speech at low bitrates [1]. However, existing models with excellent performance often suffer from two critical drawbacks: high computational complexity and non-causality, rendering them unsuitable for real-time communication [2, 3]. Track 1 of the 2025 Low-Resource Audio Codec (LRAC) Challenge focuses on audio compression that balances low latency, low bitrate, and high speech quality under constrained computational resources. Track 2 further takes the interference from noise and reverberation in real-world scenarios into account.

In this paper, we introduce our two systems submitted to the challenge. The system for Track 1, named VoCodec, is an audio codec model requiring low computational resources. Based on Vocos [4], we construct VoCodec's encoder and decoder. To reduce computational overhead and latency, audio codec is performed directly in the time-frequency domain and all upsampling and downsampling operations are eliminated in the encoder and decoder. For Track 2, UL-UNAS [5], a lightweight model for speech enhancement is cascaded at the front end of VoCodec to equip our whole system with speech enhancement capability.

† Equal contribution.
* Corresponding author: `lujing@nju.edu.cn`



(a) VoCodec

(b) ConvNeXt Block

(c) ResNet Block

Fig. 1: The architecture of the proposed model

2. PROPOSED MODEL

2.1. Architecture

Based on the VQ-GANs framework [6], the architecture of the proposed VoCodec is depicted in Figure 1. Given a speech signal **x**, it is first passed through the Short-Time Fourier Transform (STFT). Then the logarithmic magnitude and phase of the complex spectrogram are concatenated along the frequency dimension. To reduce the computational complexity, a fully connected layer is employed to reduce the input's frequency dimension to 192.

The subsequent encoder follows the improved Vocos architecture in WavTokenizer[7], which consists of 6 stacked

ConvNeXt blocks and a PosNet module. 1D convolution with causal padding is used and the inverted structure from ConvNeXt is retained with an intermediate dimension of 216. A kernel size of 7 is employed for the depth-wise convolutional layers. The PosNet incorporates 4 basic ResNet blocks using a kernel size of 3 and a causal self-attention block is added after the second ResNet block. For the decoder, it is almost a mirror-symmetric structure of the encoder. However, to constrain the receiver-side's computational complexity, the inverted design from the ConvNeXt is removed, and the number of groups in the convolutions of the ResNet blocks is set to 2.

VoCodec's quantizer uses the Residual Vector Quantization (RVQ) strategy [8]. Following the improved RVQ proposed in DAC [2], the quantizer of our model is applied with 6 layers, each containing 1024 codewords. With an encoder frame rate of 100 Hz, this corresponds to 1 kbps per layer, and 6 kbps in total.

Since we perform audio codec in the time-frequency domain, intuitively, the multi-scale STFT discriminator [9] can further improve the quality of the output audio. A set of window lengths [128, 256, 512, 1024, 2048] is used, and the hop length is fixed to the window length / 4. Moreover, only this discriminator is employed throughout the training process, while other types (e.g. MSD and MPD) are not used.

2.2. Loss Functions

When training UL-UNAS, we apply the negative scale invariant SNR (SI-SNR) [10] loss and the power-compressed specturm loss as the loss functions.

For VoCodec, the generator loss $L_{generator}$ comprises three components: the multi-scale mel-spectrogram L1 loss [2] as the reconstruction loss L_{rec}, the adversarial loss L_g with the L1 feature matching loss L_{feat} involved, the same codebook L_{code} and commitment loss L_c in VQ-VAE [11] for codebook updates. The discriminator is trained separately with the adversarial loss L_d. Formally,

$$L_{rec} = \left\| \mathcal{M}\left(x\right) - \mathcal{M}\left(\hat{x}\right) \right\|_1 \tag{1}$$

$$L_g = \| 1 - D(\hat{x}) \|_2^2 \tag{2}$$

$$L_{feat} = 2 \sum_l \left\| D^l(x) - D^l(\hat{x}) \right\|_1 \tag{3}$$

$$\begin{aligned} L_{\text{generator}} = & \lambda_{\text{rec}} L_{\text{rec}} + \lambda_g L_g + \lambda_{\text{feat}} L_{\text{feat}} \\ & + \lambda_{\text{code}} \underbrace{\| \text{sg}[\mathbf{z}_e] - \mathbf{e}_k \|_2^2}_{L_{\text{code}}} + \lambda_c \underbrace{\| \mathbf{z}_e - \text{sg}[\mathbf{e}_k] \|_2^2}_{L_c} \end{aligned} \tag{4}$$

$$L_d = \| 1 - D(x) \|_2^2 + \| D(\hat{x}) \|_2^2 \tag{5}$$

where x and $\hat{x}$ denote the target and reconstructed speech, respectively, $\mathcal{M}(\cdot)$ is the mel-spectrogram transform, $D(\cdot)$ is the discriminator output, $D^l(\cdot)$ represents the feature map of the l-th discriminator layer, $\mathbf{z}_e$ is the quantizer output, and $\mathbf{e}_k$ is the codebook vector. During training, the mel-spectrograms are computed with multiple window lengths of [32, 64, 128, 256, 512, 1024, 2048] and a fixed hop length set to window length / 4. Meanwhile, different mel bin sizes of [5, 10, 20, 40, 80, 160, 320] are employed.

Finally, we assign loss weights of 15.0 for the multi-scale mel-spectrogram loss, 1.0 for the feature matching loss, 2.0 for the adversarial loss, and 1.0, 0.25 for the codebook and commitment loss, respectively.

Table 1: Latency and computational complexity of the Track 1 baseline system.

	Transmit Side		Receive Side	Overall
	Encoder	RVQ	Decoder	
Buffering Latency (ms)	10	0	0	10
Algorithmic Latency (ms)	0	0	20	20
Compute Complexity (MMACs)	194.56	1.96	144.82	349.29

Table 2: Latency and computational complexity of the system for Track 2.

	Transmit Side			Receive Side	Overall
	SE	Encoder	RVQ	Decoder	
Buffering Latency (ms)	0	10	0	0	10
Algorithmic Latency (ms)	20	0	0	20	40
Compute Complexity (MMACs)	935.36	194.56	1.96	144.82	1284.66

2.3. Two Training Stages

We first train the speech enhancement network UL-UNAS and the codec model VoCodec independently. After their training is completed, we use UL-UNAS as the front end to perform noise suppression and dereverberation on the input audio. Subsequently, we freeze the parameters of UL-UNAS and only update the parameters of the entire codec model to enable it to compensate for the spectral distortion caused by the enhancement network.

3. EXPERIMENTAL AND RESULTS

3.1. Training Data Preparation

All training data follow the cleaning and preprocessing procedures defined in the baseline of the Challenge[1]. Considering the attention module in the codec model, we extract 3-second speech segments for training VoCodec.

During the training process of UL-UNAS and the final cascade system, each speech sample is combined with background noise, where signal-to-noise ratio (SNR) is uniformly distributed between -5 dB and 30 dB. In addition, reverberation is randomly introduced, and the final training target is speech signals with early reverberation.

3.2. Implementation Details

STFT is computed using a square root Hanning window of a length of 30 ms, a hop length of 10 ms, and an FFT length of 720, resulting in a buffering latency of 10 ms and an algorithmic latency of 20 ms due to the inverse STFT processing.

[1]https://github.com/cisco-open/lrac_data_generation

Table 3: Performance Comparison on the Open Test Set for Track 1

Bitrate	Model	Condition	ScoreQ-ref ↓	UTMOS ↑	Sheet-SSQA ↑	PESQ ↑	Audiobox AE-CE ↑
6 kbps	Baseline	Clean	0.35	3.23	3.84	2.67	5.28
		Noisy	0.82	2.76	3.12	1.81	4.37
		Reverb	1.13	1.32	2.22	1.18	3.43
	VoCodec	Clean	**0.17**	**3.73**	**4.22**	**3.20**	**5.66**
		Noisy	**0.70**	**3.10**	**3.43**	**2.03**	**4.82**
		Reverb	**0.94**	**1.55**	**2.80**	**1.21**	**3.98**
1 kbps	Baseline	Clean	1.15	1.44	1.84	1.15	3.90
		Noisy	1.29	1.33	1.72	1.11	3.40
		Reverb	1.36	1.26	1.85	1.07	2.94
	VoCodec	Clean	**0.40**	**3.24**	**3.55**	**1.95**	**5.31**
		Noisy	**0.83**	**2.67**	**2.93**	**1.56**	**4.43**
		Reverb	**1.10**	**1.48**	**2.19**	**1.17**	**3.59**

Table 4: Performance Comparison on the Open Test Set for Track 2

Bitrate	Model	Condition	ScoreQ-ref ↓	UTMOS ↑	Sheet-SSQA ↑	PESQ ↑	Audiobox AE-CE ↑
6 kbps	Baseline	Clean	0.43	2.97	3.55	2.13	2.97
		Noisy	0.75	2.56	2.92	1.73	4.60
		Reverb	0.92	1.79	2.67	1.29	4.25
	VoCodec	Clean	**0.18**	**3.74**	**4.21**	**3.06**	**5.68**
		Noisy	**0.50**	**3.26**	**3.62**	**2.19**	**5.00**
		Reverb	**0.88**	**2.02**	**2.70**	**1.38**	4.20
1 kbps	Baseline	Clean	1.01	1.37	2.07	1.21	3.96
		Noisy	1.15	1.35	1.95	1.18	3.70
		Reverb	1.12	1.32	2.43	1.15	3.55
	VoCodec	Clean	**0.41**	**3.21**	**3.50**	**1.92**	**5.29**
		Noisy	**0.68**	**2.81**	**3.00**	**1.63**	**4.75**
		Reverb	**1.04**	**1.75**	2.20	**1.26**	**3.95**

As shown in Table 1, our proposed VoCodec comprises 3.47 M parameters and has a computational complexity of 349.29 MMAC/s [2], with the receiver-side computation accounting for only 144.82 MMAC/s. In Track 2, we adopt the same network architecture from UL-UNAS [5], and scale up the number of intermediate channels to [48, 96, 108, 108, 64], which results in a computational complexity of 935.36 MMAC/s. The whole system has a computational complexity of 1.28 GMAC/s with 5.34 M parameters shown in Table 2. The total latency is 50 ms, of which 20 ms is attributed to the additional look-ahead in UL-UNAS.

We train UL-UNAS and VoCodec independently on 8 NVIDIA RTX 4090 GPUs. The batch size for UL-UNAS is set to 4 per GPU, while that for VoCodec is 24 per GPU. UL-UNAS and VoCodec are trained for 400 and 1000 epochs, with 1250 and 500 iterations per epoch, respectively. During training, we use the AdamW optimizer [12] and employ a linear warmup scheduler followed by cosine annealing. In the joint training phase, we adopt the same configuration as used in the training of VoCodec, and conduct the training process for a total of 500 epochs.

Meanwhile, we employ a systematic strategy to select the final checkpoint of the model. The validation objective metrics are evaluated at regular intervals during training and the checkpoint with the best performance is selected.

3.3. Results

Evaluation on the test set is conducted using the official metrics provided by the Challenge. Scoreq-ref [13], UTMOS [14], Audiobox AE-CE [15], PESQ [16], and Sheet-SSQA [17, 18] are selected to compare the quality and naturalness of speech recovered by the decoder of the codec.

The experimental results are summarized in Table 3 and Table 4. It can be seen that our two systems outperform the official baseline models[3] across all metrics, particularly on the

[2]The computational complexity is calculated by ptflops: `https://github.com/tel-0s/ptflops`.

[3]`https://github.com/cisco-open/espnet/tree/master/egs2/lrac`

clean and noisy test sets.

4. CONCLUSION

This paper introduces VoCodec, an efficient lightweight speech codec, and our two systems in the LRAC 2025 Challenge. Experiments show that our systems achieve superior performance over the baseline models.

5. REFERENCES

[1] Yiwei Guo, Zhihan Li, Hankun Wang, Bohan Li, Chongtian Shao, Hanglei Zhang, Chenpeng Du, Xie Chen, Shujie Liu, and Kai Yu, "Recent advances in discrete speech tokens: A review," *CoRR*, vol. abs/2502.06490, February 2025.

[2] Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar, "High-fidelity audio compression with improved RVQGAN," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[3] Detai Xin, Xu Tan, Shinnosuke Takamichi, and Hiroshi Saruwatari, "Bigcodec: Pushing the limits of low-bitrate neural speech codec," *arXiv preprint arXiv:2409.05377*, 2024.

[4] Hubert Siuzdak, "Vocos: Closing the gap between time-domain and fourier-based neural vocoders for high-quality audio synthesis," *arXiv preprint arXiv:2306.00814*, 2023.

[5] Xiaobin Rong, Dahan Wang, Yuxiang Hu, Changbao Zhu, Kai Chen, and Jing Lu, "Ul-unas: Ultra-lightweight u-nets for real-time speech enhancement via network architecture search," 2025.

[6] Patrick Esser, Robin Rombach, and Bjorn Ommer, "Taming transformers for high-resolution image synthesis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 12873–12883.

[7] Shengpeng Ji, Ziyue Jiang, et al., "Wavtokenizer: an efficient acoustic discrete codec tokenizer for audio language modeling," *arXiv preprint arXiv:2408.16532*, 2024.

[8] Biing-Hwang Juang and A. Gray, "Multiple stage vector quantization for speech coding," in *ICASSP '82. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1982, vol. 7, pp. 597–600.

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

[10] Jonathan Le Roux, Scott Wisdom, Hakan Erdogan, and John R. Hershey, "Sdr – half-baked or well done?," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 626–630.

[11] Aaron van den Oord, Oriol Vinyals, and koray kavukcuoglu, "Neural discrete representation learning," in *Advances in Neural Information Processing Systems*, I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. 2017, vol. 30, Curran Associates, Inc.

[12] Ilya Loshchilov and Frank Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2017.

[13] Alessandro Ragano, Jan Skoglund, and Andrew Hines, "Scoreq: Speech quality assessment with contrastive regression," in *Advances in Neural Information Processing Systems*. 2024, vol. 37, pp. 105702–105729, Curran Associates, Inc.

[14] Takaaki Saeki and Detai Xin and Wataru Nakata and Tomoki Koriyama and Shinnosuke Takamichi and Hiroshi Saruwatari, "UTMOS: UTokyo-SaruLab System for VoiceMOS Challenge 2022," in *Interspeech 2022*, 2022, pp. 4521–4525.

[15] Andros Tjandra, Yi-Chiao Wu, Baishan Guo, John Hoffman, Brian Ellis, Apoorv Vyas, Bowen Shi, Sanyuan Chen, Matt Le, Nick Zacharov, Carleigh Wood, Ann Lee, and Wei-Ning Hsu, "Meta audiobox aesthetics: Unified automatic quality assessment for speech, music, and sound," 2025.

[16] A.W. Rix, J.G. Beerends, M.P. Hollier, and A.P. Hekstra, "Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs," in *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*, 2001, vol. 2, pp. 749–752 vol.2.

[17] Wen-Chin Huang, Erica Cooper, and Tomoki Toda, "SHEET: A Multi-purpose Open-source Speech Human Evaluation Estimation Toolkit," in *Proc. Interspeech*, 2025, pp. 2355–2359.

[18] Wen-Chin Huang, Erica Cooper, and Tomoki Toda, "Mos-bench: Benchmarking generalization abilities of subjective speech quality assessment models," 2024.