Interspeech 2024
1-5 September 2024, Kos, Greece

Towards Expressive Zero-Shot Speech Synthesis with Hierarchical Prosody Modeling

Yuepeng Jiang[1], *Tao Li*[1], *Fengyu Yang*[2], *Lei Xie*[1*], *Meng Meng*[2], *Yujun Wang*[2]

[1]Audio, Speech and Language Processing Group (ASLP@NPU), School of Software, Northwestern Polytechnical University, Xi'an, China
[2]Xiaomi AI Lab, Beijing, China
jiangyp@mail.nwpu.edu.cn, lxie@nwpu.edu.cn

Abstract

Recent research in zero-shot speech synthesis has made significant progress in speaker similarity. However, current efforts focus on timbre generalization rather than prosody modeling, which results in limited naturalness and expressiveness. To address this, we introduce a novel speech synthesis model trained on large-scale datasets, including both timbre and hierarchical prosody modeling. As timbre is a global attribute closely linked to expressiveness, we adopt a global vector to model speaker timbre while guiding prosody modeling. Besides, given that prosody contains both global consistency and local variations, we introduce a diffusion model as the pitch predictor and employ a prosody adaptor to model prosody hierarchically, further enhancing the prosody quality of the synthesized speech. Experimental results show that our model not only maintains comparable timbre quality to the baseline but also exhibits better naturalness and expressiveness. The synthesized samples can be found at: https://rxy-j.github.io/HPMD-TTS/
Index Terms: speech synthesis, zero-shot, prosody modeling, denoising diffusion probabilistic model

1. Introduction

The aim of text-to-speech (TTS) synthesis is to convert text into human-like speech. With the development of neural networks and deep learning, the naturalness and intelligibility of the synthesized speech for speakers *seen* in the training corpus have been promoted to a human-like level [1, 2, 3, 4, 5]. However, such systems can only generate voice for speaker-limited recording-studio datasets, which limits the diversity of generated speaker timbre, prosody, and style [6]. Therefore, how to generate speech for any speaker who was *unseen* during the TTS model training, solely relying on a single acoustic reference of that speaker, i.e. zero-shot speech synthesis, is gaining growing attention recently [7, 8, 9].

In zero-shot speech synthesis, the primary challenge lies in enhancing the robustness and generalizability of speaker timbre modeling [10], thereby guaranteeing a precise reproduction of the unseen target speaker's timbre in synthesized speech. In prior studies, a commonly employed speaker modeling approach has been to leverage a speaker encoder to extract speaker embeddings from the reference speech, subsequently as a speaker timbre control condition in the acoustic model. Based on training methods, the current speaker encoders can be roughly classified into jointly trained [11, 12] and pre-trained methods [7, 13, 14]. For jointly trained methods, researchers focus on training an encoder network jointly with acoustic models, like using unsupervised latent embedding technologies [15, 16], such as global speaker embeddings (GSEs) [17] or variational autoencoder (VAE) [18]. However, these methods are plagued by unsatisfactory generalization performance due to training data's scale limitations, resulting in poor speaker similarity. Pre-trained methods draw on transfer learning, where speaker encoders are independently trained based on other tasks, e.g., speaker verification or speech recognition. In this case, the speaker encoder utilizes large amounts of data for training and can provide more robust representations of speaker characteristics. ZSM-SS [19] constructs a speaker encoder based on Wav2vec 2.0 [20] and achieves good performance in speaker similarity. YourTTS [9] adopts an encoder pre-trained based on speaker recognition and designs a speaker consistency loss function to optimize model training, achieving state-of-the-art results in zero-shot testing on the VCTK dataset. Despite these advances, the generalization capability of these methods is still insufficient, and the naturalness and timbre similarity are still not ideal in the case of synthesizing speech for unseen speakers.

Recently, zero-shot TTS methods [21, 6, 22] that rely on large-scale corpora have gained increasing attention. The diversity of data available in such corpora has significantly enhanced the generalization capabilities of TTS models, enabling them to capture the diverse speaker timbre, prosody, and style, effectively remarkable speaker similarity of unseen speakers. Based on their data representation methods, these methods can be classified into two types: discrete tokens and continuous vectors. VALL-E [21] uses a neural codec based on Residual Vector Quantization (RVQ) to quantize speech into tokens, taking both text and audio tokens as input, while a language model is used to generate the target speech tokens autoregressively. SPEAR-TTS [23], modifies AudioLM [24] and achieves a stepwise transformation from text to semantic tokens and then to acoustic tokens. NaturalSpeech2 [6] uses a neural codec based on RVQ to encode audio into continuous vectors as its data representation. Then, a latent diffusion model [25] is adopted to predict these continuous vectors. However, these methods neglect explicit modeling of prosody, which could result in insufficient naturalness and expressiveness of the generated speech, especially when the reference speech is already highly expressive and varied in prosody.

To harness the diversity of large-scale corpora and simultaneously enhance the prosody modeling capabilities of the model, we introduce a novel zero-shot speech synthesis model that incorporates a diffusion model-based prosody predictor and a hierarchical prosody adaptor. First, considering that speaker timbre is a global attribute that does not vary over time, we adopt a speaker encoder to extract global speaker embedding. As quantized audio excessively discards speech details, it fails to guide speaker timbre restoration. Therefore, we employ mel

* Corresponding author.

10.21437/Interspeech.2024-2506

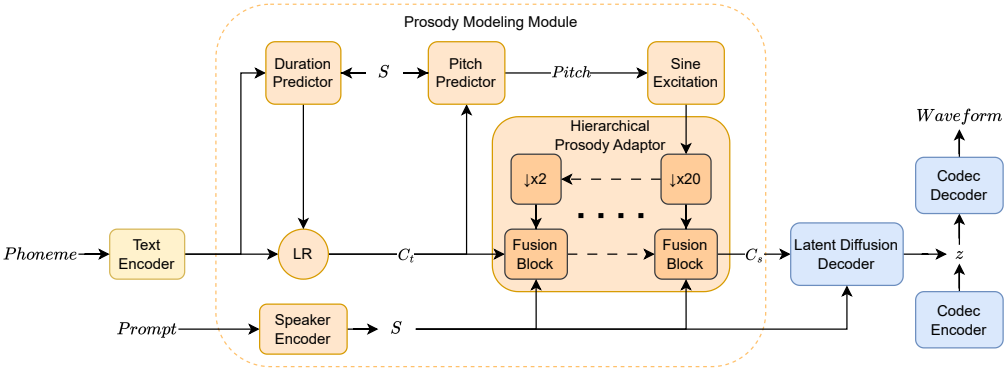


Figure 1: *Architecture overview of the proposed model*

spectrograms as inputs to the speaker encoder. Second, we employ a diffusion model as a pitch predictor to match speech prosody diversity by leveraging its natural advantage in generating content diversity. Finally, since prosody shows both global consistency and local variations, we employ a prosody adaptor to model prosody hierarchically, such as frame-level, phoneme-level, and word-level, to improve the prosody performance of synthesized speech. Experimental results demonstrate that our model achieves better naturalness while maintaining speaker similarity compared to the comparison models. The results also prove that utilizing a hierarchical modeling approach significantly improves prosody details and jitter issues compared to directly adding prosody information to intermediate variables.

2. Proposed Approach

2.1. Overview

As shown in Figure 1, the proposed model consists of a text encoder, a prosody modeling module, and a latent diffusion decoder. The prosody modeling module consists of a speaker encoder, a diffusion-based pitch predictor, a duration predictor, and a hierarchical prosody adaptor. The diffusion-based pitch predictor uses the text encoder output C_t and the global speaker representation S as conditions to predict the fundamental frequency. The hierarchical prosody adaptor uses the fundamental frequency for hierarchical prosody modeling. Meanwhile, guided by the global speaker representation S, it performs speaker timbre modeling, transforming the text encoder output C_t into a content representation containing prosody and timbre information C_s. Then, the latent diffusion decoder, conditioned on C_s and S, generates the latent vector z and further enhances the timbre information in z. Finally, a waveform is reconstructed from z by the codec decoder.

2.2. Pitch Predictor Based on Diffusion

Given the diversity of speech prosodic patterns, we adopted a diffusion model as a predictor to leverage its natural advantage in generating content diversity. The denoising diffusion probabilistic model (DDPM) is a generative model that uses a predefined forward process to gradually transform data into pure noise. Subsequently, the model learns the reverse operation of this process, through the denoising process, it gradually reconstructs the original data from the noise. Its forward process can be represented using the following equation:

$$q\left(\mathbf{x}_t \mid \mathbf{x}_{t-1}\right) := \mathcal{N}\left(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t \mathbf{I}\right). \quad (1)$$

At each step of the diffusion process, a very tiny Gaussian noise is added to x_{t-1} at the current time $t-1$ to obtain x_t. This Gaussian noise is determined by variance schedule $\beta = \{\beta_1, \cdots, \beta_T\}$. The complete forward process can be described as follows:

$$q\left(\mathbf{x}_{1:T} \mid \mathbf{x}_0\right) := \prod_{t=1}^{T} q\left(\mathbf{x}_t \mid \mathbf{x}_{t-1}\right), \quad (2)$$

where T is the number of steps when the diffusion process terminates. When T is large enough, x_T can be regarded as random noise following the Gaussian distribution $N(0, I)$. Therefore, we can let the reverse process start at $x_T \sim N(0, I)$. For each step of the reverse process, it can be described by the following equation:

$$p_\theta\left(\mathbf{x}_{t-1} \mid \mathbf{x}_t\right) := \mathcal{N}\left(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta\left(\mathbf{x}_t, t\right), \sigma_t^2 \mathbf{I}\right), \quad (3)$$

where $\sigma^2 = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$ $\alpha_t := 1 - \beta_t$ and $\bar{\alpha} := \prod_{s=1}^{t} \alpha_s$. θ is a learnable parameter. In general, there is no direct way to obtain the forward process $q(x_t|x_{t-1})$ corresponding to the reverse process at time t. Therefore, we use a neural network to predict θ, allowing it to approximate the forward process at time t. To train parameter θ, we minimize the variational bound of negative log-likelihood. [26] has simplified it:

$$L(\theta) := \mathbb{E}_{t, \mathbf{x}_0, \boldsymbol{\epsilon}}\left[\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta\left(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t\right)\right\|^2\right], \quad (4)$$

where sample $\epsilon \sim N(0, I)$ and $\epsilon(\cdot)$ is the outputs of the neural network. The complete proof of denoising diffusion probabilistic models can be found in [26].

For the model we are using, the denoiser predicts θ by using C_t and S as conditions. Therefore, in this model, the actual denoising process is described by the following equation:

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta\left(\mathbf{x}_t, C_t, s, t\right)\right) + \sigma_t \mathbf{z}. \quad (5)$$

Moreover, we apply gradient clipping to its inputs. Therefore, the predictor in this model is a completely independent module.

2.3. Hierarchical Prosody Adaptor

Considering that the global consistency and local variability in prosody, we adopt a prosody adaptor to model the prosody in a hierarchical manner. The detailed structure of the hierarchical prosody adaptor is illustrated in Figure 1. This module takes

Table 1: *Evaluations for different models with 95% confidence intervals.*

Method	Emotional				Overall			
	NMOS ↑	**SMOS** ↑	**CER(%)** ↓	**SECS** ↑	**NMOS** ↑	**SMOS** ↑	**CER(%)** ↓	**SECS** ↑
Proposed	**3.65**±0.14	**3.50**±0.15	**6.35**	0.818	**3.62**±0.10	3.54±0.11	**5.99**	**0.829**
NS2	3.50±0.14	3.30±0.17	7.43	0.807	3.54±0.13	3.36±0.15	6.56	0.815
VALL-E	3.52±0.16	3.49±0.22	13.07	**0.820**	3.60±0.13	**3.57**±0.17	10.18	0.826

the content information C_t, frame-level pitch, and speaker embedding S as inputs and outputs C_s containing prosody, timbre, and content information. Inspired by [27][28], we utilize sample-level sinusoidal excitation signals to further represent pitch information. Firstly, we need to expand the frame-level f0 to the sample-level f0 $f_0[n]$ through linear interpolation. Subsequently, the corresponding sine excitation signals $p[n]$ can be generated by the following two formulas:

$$p[n] = \begin{cases} \sum_{k=1}^{K} \sin\left(\frac{2\pi k \sum_{i=0}^{n} f_0[i]}{f_s}\right), & f_0[n] > 0 \\ 0, & f_0[n] = 0 \end{cases} \quad (6)$$

$$K = min(K_{max}, \left\lfloor \frac{f_s}{2f_0[n]} \right\rfloor), \quad (7)$$

where $k \in \{1, 2, ..., K\}$ and n are the indexes of harmonics and time, respectively. K is the largest number of harmonics that can be achieved corresponding to the $f_0[n]$. f_s is the audio sampling rate. In this paper, we set K_{max} to 200, which limits the largest harmonics index to 200.

By downsampling the sinusoidal excitation signal multiple times, we can obtain information at different scales. Unlike the method proposed in [27], it is not required for the length of the sinusoidal signal to match the length of the content information in our module. Therefore, we increase the downsampling factor to ensure that the scale difference between the two signals is maximized. During the process of multiple downsampling, the scale of the signal can approximate frame-level, phoneme-level, and word-level, respectively. This provides hierarchical prosody information and facilitates modeling at different scales.

In the system we propose, there are a total of four downsampling layers and corresponding modeling units. We set downsampling factors to [20, 10, 10, 2], respectively. During the data preprocessing process, the hop size is set to 200 when extracting the fundamental frequency, which means that the data needs to be downsampled by 200 times from the sample-level to the frame-level. To prevent the size of a single downsampling layer from being too large, we achieve 200 times downsampling through the first two downsampling layers. According to research [29], the average speaking rate in Chinese is approximately 150 to 200 characters per minute. Considering that our phoneme set comprises an average of two phonemes per character, and accounting for factors such as speech pauses, the ratio between frame-level and phoneme-level is approximately 10 to 15. Therefore, we set the downsampling factor for the third level to 10 and for the fourth level to 2, corresponding to the phoneme-level and word-level, respectively. Subsequently, we employ cross-attention at each scale to fuse prosody information.

Additionally, to better integrate the global speaker timbre and prosody information S, we introduce the SALN module for information injection, ensuring that each prosody modeling unit can observe global and fine-grained prosody information at the same time. This approach not only preserves the consistency of global features but also enhances the system's adaptability to different speakers, further strengthening the model's ability in speech modeling.

3. Experiments

3.1. Dataset and Preprocessing

The WenetSpeech4TTS [30] dataset is used in experiments. Considering that it is originally a Mandarin corpus designed for speech recognition, we process it to make it applicable for TTS. The annotated data is re-segmented, then denoised and filtered to exclude speech with multiple speakers. The data is then scored using DNSMOS [31], and we obtain about 8000 hours of data with scores above 3.6 and about 4000 hours with scores above 3.8, denoted as Large (*L*) and Medium (*M*), respectively. These two sets of data are used for subsequent model training. The data is resampled to 16k and the loudness is normalized.

To validate the zero-shot voice cloning capability, prosody modeling ability, and expressiveness of our proposed model, we select 15 speakers for evaluation. Specifically, these speakers are chosen from AISHELL3[1], an internal emotional dataset [32], and WenetSpeech's evaluation set, with 5 speakers selected from each source. The internal emotion dataset comprises six typical emotions i.e., happy, anger, sad, fear, surprise, and disgust. By utilizing speakers selected from this dataset, we can evaluate the naturalness performance of the proposed model when presented with highly expressive references.

We process the text using pypinyin[2], extracting phonemes and tones as model inputs. In addition, the mel spectrogram is extracted from the original waveforms and randomly samples 1 to 3 seconds of segments as a prompt based on the audio length. The fundamental frequency is extracted using PyWORLD[3] toolkit, and we apply interpolation for smoothing. We utilize Encodec [33] trained on the WenetSpeech dataset as the model's acoustic codec to extract z for training. While maintaining the codebook size at 1024, we increase the codebook dimension from 128 to 256. We use a Kaldi[4] model trained on a 2000-hour subset of the set *L* to obtain external duration information.

3.2. Training

For the sinusoidal excitation signal extraction, we set K_{max} to 200. The diffusion steps for the pitch predictor and latent diffusion decoder are set to 100 and 200, respectively. We utilize AdamW optimizer for training, with β_1 set to 0.8, β_2 set to 0.99, and weight decay λ set to 0.01. The initial learning rate is set to 1e-4. Our model is trained up to 600k training steps on 4 NVIDIA A6000 GPUs with a batch size of 16. We first train the model on the set *L* and then fine-tune it on the set *M*.

[1] https://www.aishelltech.com/aishell_3

[2] https://github.com/mozillazg/python-pinyin

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

[4] https://github.com/kaldi-asr/kaldi

3.3. Baseline

We compare our proposed model with NaturalSpeech2 (NS2) [6] and VALL-E [21]. We use the NS2 and VALL-E code provided by the Amphion [34] toolkit to build these models, using their default configurations. The training strategy for both models is similar to ours.

4. Results

4.1. Objective Evaluation

We evaluate the model performance on a reserved testset of 150 sentences using Character Error Rate (CER) and speaker embedding cosine similarity (SECS). To reduce the error rate caused by homophone transcription mistakes, we carefully select texts that do not contain proper nouns, names, and other error-prone terms to compose this testset. We use Paraformer [35] pre-trained on 60,000 hours of Mandarin Chinese to transcribe the generated speech and calculate CER. For SECS, we use Resemblyzer [5] to extract speaker embedding and calculate cosine similarity using the function build in pytorch. For Paraformer and Resemblyzer, we utilize the official code and pre-trained checkpoints. Objective test results are shown in the table 1. For comparison, the test results of the speakers from the emotional dataset are listed separately.

As shown in table 1, our model achieves the lowest CER, further reducing it by approximately 0.5% compared to NS2, which is also a non-autoregressive model. The NS2 training process involves extracting prompts from the original audio based on phoneme boundaries and then synthesizing the remaining parts, such a procedure is extremely dependent on the accuracy of external duration information. Specifically, the data quality of WenetSpeech is relatively lower than recording-studio benchmarking datasets (e.g., DiDiSpeech [36]), resulting in lower duration accuracy of WenetSpeech data obtained through Kaldi, which could be the main factor affecting the CER of NS2. Besides, VALL-E has significantly higher CER due to autoregressive model instability, misreadings, and incorrect pauses. For the SECS test, there were no significant differences in the scores of the three models, whether analyzing overall performance or the emotional dataset.

4.2. Subjective Evaluation

We conduct Mean Opinion Score (MOS) tests to evaluate the speaker similarity and naturalness of synthesized speech, denoted as SMOS and NMOS, respectively. Each MOS test involves 20 listeners who rate the speech on a scale from 1 to 5 based on subjective perception. The results of the subjective test are also shown in the table 1.

Through comparison, it can be found that the proposed model outperforms NS2 and VALL-E in both speaker similarity and naturalness. Specifically, when dealing with speakers from the emotional dataset, the naturalness of the proposed model does not significantly decline, while VALL-E and NS2 show varying degrees of naturalness degradation, especially VALL-E, which shows the most notable decline. Furthermore, the listening tests also show that the audio quality of VALL-E and NS2 decreases significantly when dealing with speakers from the emotional dataset, which may be an important reason for their poor performance in speaker similarity and naturalness.

[5] https://github.com/resemble-ai/Resemblyzer

Table 2: *Results of ablation studies.*

Model	NMOS (↑)
Proposed	3.62±0.10
- pitch predictor based on diffusion	3.58±0.07
- hierarchical prosody modeling	3.49±0.16

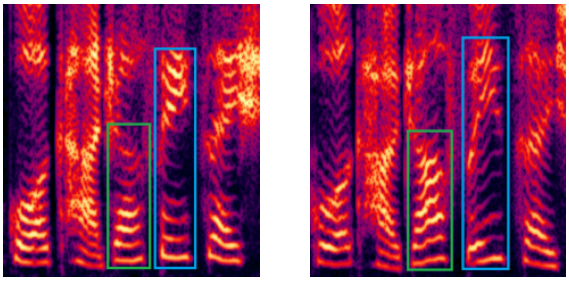


Figure 2: *The spectrograms of synthesized samples in hierarchical prosody modeling ablation.*

4.3. Ablation Study

To validate the effectiveness of our model structure, we carry out a series of ablation studies. We conduct ablation experiments on the pitch predictor and hierarchical prosody modeling. In the ablation study for the pitch predictor, we replace our model's predictor with the pitch predictor from the FastSpeech [2]. For the hierarchical prosody modeling ablation, we remove the related structures. We directly encode frame-level fundamental frequency information and then add it to C_t.

The experimental results show that removing the fundamental frequency predictor leads to a slight decrease in naturalness, indicating that the introduction of diffusion does indeed play a positive role in enhancing prosody diversity. The significant decrease in naturalness after removing hierarchical prosody modeling highlights the importance of implementing prosody modeling at different scales. Furthermore, in comparative listening tests, we find that the direct addition of fundamental frequency information to C_t causes noticeable pronunciation jitter, shown in Figure 2, which makes the speech sound unnatural.

5. Conclusions

To leverage the data diversity of large-scale corpora and simultaneously enhance the prosody modeling performance, this paper proposes a novel zero-shot speech synthesis model that incorporates a diffusion model-based prosody predictor and a hierarchical prosody adaptor. Specifically, we employ global vectors to represent speaker timbre and utilize them as a guide for prosody modeling as the expressiveness is correlated to the speaker. Considering that prosody is reflected in both global consistency and local variations, a diffusion model is introduced as the pitch predictor while a prosody adaptor is adopted to model hierarchical prosody variance, thus enhancing the prosody performance of the synthesized speech. Experimental results show that our model not only maintains comparable timbre quality to the baseline but also exhibits better naturalness and expressiveness.

6. References

[1] Y. Wang, R. J. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. V. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: Towards end-to-end speech synthesis," in *Proc. Interspeech*, 2017, pp. 4006–4010.

[2] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T. Liu, "Fastspeech: Fast, robust and controllable text to speech," in *Proc. NeurIPS*, 2019, pp. 3165–3174.

[3] V. Popov, I. Vovk, V. Gogoryan, T. Sadekova, and M. A. Kudinov, "Grad-tts: A diffusion probabilistic model for text-to-speech," in *Proc. ICML*, M. Meila and T. Zhang, Eds., vol. 139, 2021, pp. 8599–8608.

[4] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," in *Proc. ICML*, 2021, pp. 5530–5540.

[5] Y. Liu, Z. Xu, G. Wang, K. Chen, B. Li, X. Tan, J. Li, L. He, and S. Zhao, "Delightfultts: The microsoft speech synthesis system for blizzard challenge 2021," *CoRR*, vol. abs/2110.12612, 2021.

[6] K. Shen, Z. Ju, X. Tan, Y. Liu, Y. Leng, L. He, T. Qin, S. Zhao, and J. Bian, "Naturalspeech 2: Latent diffusion models are natural and zero-shot speech and singing synthesizers," *CoRR*, vol. abs/2304.09116, 2023.

[7] S. Arik, J. Chen, K. Peng, W. Ping, and Y. Zhou, "Neural voice cloning with a few samples," *Advances in neural information processing systems*, vol. 31, 2018.

[8] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, P. Nguyen, R. Pang, I. Lopez Moreno, Y. Wu *et al.*, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," *Advances in neural information processing systems*, vol. 31, 2018.

[9] E. Casanova, J. Weber, C. D. Shulby, A. C. Júnior, E. Gölge, and M. A. Ponti, "Yourtts: Towards zero-shot multi-speaker TTS and zero-shot voice conversion for everyone," in *Proc. ICML*, 2022, pp. 2709–2720.

[10] Y. Deng, L. He, and F. K. Soong, "Modeling multi-speaker latent space to improve neural TTS: quick enrolling new speaker and enhancing premium voice," *CoRR*, vol. abs/1812.05253, 2018.

[11] M. Kang, D. Min, and S. J. Hwang, "Grad-stylespeech: Any-speaker adaptive text-to-speech synthesis with diffusion models," in *Proc. ICASSP*, 2023, pp. 1–5.

[12] D. Min, D. B. Lee, E. Yang, and S. J. Hwang, "Meta-stylespeech: Multi-speaker adaptive text-to-speech generation," in *Proc. ICML*, 2021, pp. 7748–7759.

[13] E. Cooper, C.-I. Lai, Y. Yasuda, F. Fang, X. Wang, N. Chen, and J. Yamagishi, "Zero-shot multi-speaker text-to-speech with state-of-the-art neural speaker embeddings," in *Proc. ICASSP*, 2020, pp. 6184–6188.

[14] Y. Zhang, H. Che, J. Li, C. Li, X. Wang, and Z. Wang, "One-shot voice conversion based on speaker aware module," in *Proc. ICASSP*, 2021, pp. 5959–5963.

[15] C. Zhang, Y. Zhu, and S. Zhu, "Metastyle: Three-way trade-off among speed, flexibility, and quality in neural style transfer," in *Proc. AAAI*, 2019, pp. 1254–1261.

[16] M. Kang, D. Min, and S. J. Hwang, "Grad-stylespeech: Any-speaker adaptive text-to-speech synthesis with diffusion models," in *Proc. ICASSP*, 2023, pp. 1–5.

[17] H. Lu, Z. Wu, D. Dai, R. Li, S. Kang, J. Jia, and H. Meng, "One-shot voice conversion with global speaker embeddings." in *Proc. Interspeech*, 2019, pp. 669–673.

[18] W. Hsu, Y. Zhang, R. J. Weiss, H. Zen, Y. Wu, Y. Wang, Y. Cao, Y. Jia, Z. Chen, J. Shen, P. Nguyen, and R. Pang, "Hierarchical generative modeling for controllable speech synthesis," in *Proc. ICLR*, 2019.

[19] N. Kumar, S. Goel, A. Narang, and B. Lall, "Normalization driven zero-shot multi-speaker speech synthesis." in *Proc. Interspeech*, 2021, pp. 1354–1358.

[20] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Proc. NeurIPS*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020.

[21] C. Wang, S. Chen, Y. Wu, Z. Zhang, L. Zhou, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li, L. He, S. Zhao, and F. Wei, "Neural codec language models are zero-shot text to speech synthesizers," *CoRR*, vol. abs/2301.02111, 2023.

[22] Z. Jiang, Y. Ren, Z. Ye, J. Liu, C. Zhang, Q. Yang, S. Ji, R. Huang, C. Wang, X. Yin, Z. Ma, and Z. Zhao, "Mega-tts: Zero-shot text-to-speech at scale with intrinsic inductive bias," *CoRR*, vol. abs/2306.03509, 2023.

[23] E. Kharitonov, D. Vincent, Z. Borsos, R. Marinier, S. Girgin, O. Pietquin, M. Sharifi, M. Tagliasacchi, and N. Zeghidour, "Speak, read and prompt: High-fidelity text-to-speech with minimal supervision," *arXiv preprint arXiv:2302.03540*, 2023.

[24] Z. Borsos, R. Marinier, D. Vincent, E. Kharitonov, O. Pietquin, M. Sharifi, D. Roblek, O. Teboul, D. Grangier, M. Tagliasacchi *et al.*, "Audiolm: a language modeling approach to audio generation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023.

[25] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.

[26] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[27] K. Song, Y. Zhang, Y. Lei, J. Cong, H. Li, L. Xie, G. He, and J. Bai, "DSPGAN: A gan-based universal vocoder for high-fidelity TTS by time-frequency domain supervision from DSP," in *Proc. ICASSP*, 2023, pp. 1–5.

[28] Z. Ning, Y. Jiang, Z. Wang, B. Zhang, and L. Xie, "Vits-based singing voice conversion leveraging whisper and multi-scale F0 modeling," in *Proc. ASRU*, 2023, pp. 1–8.

[29] F. Pellegrino, C. Coupé, and E. Marsico, "A cross-language perspective on speech information rate," *Language*, pp. 539–558, 2011.

[30] L. Ma, D. Guo, K. Song, Y. Jiang, S. Wang, L. Xue, W. Xu, H. Zhao, B. Zhang, and L. Xie, "Wenetspeech4tts: A 12,800-hour mandarin tts corpus for large speech generation model benchmark," *CoRR*, vol. abs/2406.05763, 2024.

[31] C. K. A. Reddy, V. Gopal, and R. Cutler, "Dnsmos: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors," in *Proc. ICASSP*, 2021, pp. 6493–6497.

[32] T. Li, X. Wang, Q. Xie, Z. Wang, and L. Xie, "Cross-speaker emotion disentangling and transfer for end-to-end speech synthesis," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 1448–1460, 2022.

[33] A. Défossez, J. Copet, G. Synnaeve, and Y. Adi, "High fidelity neural audio compression," *CoRR*, vol. abs/2210.13438, 2022.

[34] X. Zhang, L. Xue, Y. Gu, Y. Wang, H. He, C. Wang, X. Chen, Z. Fang, H. Chen, J. Zhang, T. Y. Tang, L. Zou, M. Wang, J. Han, K. Chen, H. Li, and Z. Wu, "Amphion: An open-source audio, music and speech generation toolkit," *arXiv*, vol. abs/2312.09911, 2024.

[35] Z. Gao, S. Zhang, I. McLoughlin, and Z. Yan, "Paraformer: Fast and Accurate Parallel Transformer for Non-autoregressive End-to-End Speech Recognition," in *Proc. Interspeech*, 2022, pp. 2063–2067.

[36] T. Guo, C. Wen, D. Jiang, N. Luo, R. Zhang, S. Zhao, W. Li, C. Gong, W. Zou, K. Han, and X. Li, "Didispeech: A large scale mandarin speech corpus," in *Proc. ICASSP*, 2021, pp. 6968–6972.