Spotlight-TTS: Spotlighting the Style via Voiced-Aware Style Extraction and Style Direction Adjustment for Expressive Text-to-Speech

Nam-Gyu Kim, Deok-Hyeon Cho, Seung-Bin Kim, Seong-Whan Lee[†]

Department of Artificial Intelligence, Korea University, Seoul, Korea

ng_kim@korea.ac.kr, dh_cho@korea.ac.kr, sb-kim@korea.ac.kr, sw.lee@korea.ac.kr

Abstract

Recent advances in expressive text-to-speech (TTS) have introduced diverse methods based on style embedding extracted from reference speech. However, synthesizing high-quality expressive speech remains challenging. We propose Spotlight-TTS, which exclusively emphasizes style via voiced-aware style extraction and style direction adjustment. Voiced-aware style extraction focuses on voiced regions highly related to style while maintaining continuity across different speech regions to improve expressiveness. We adjust the direction of the extracted style for optimal integration into the TTS model, which improves speech quality. Experimental results demonstrate that Spotlight-TTS achieves superior performance compared to baseline models in terms of expressiveness, overall speech quality, and style transfer capability. Our audio samples are publicly available.[1]

Index Terms: Text-to-speech, expressive speech synthesis, style transfer, vector quantization

1. Introduction

Text-to-speech (TTS) [1] aims to synthesize speech from input text. With recent advancements in deep learning technology [2, 3, 4], the naturalness of synthesized speech has improved significantly [5, 6]. Despite the development of general TTS systems, synthesizing human-like speech for applications such as virtual assistants and audiobooks remains challenging due to the lack of expressiveness of synthesized speech. To address this limitation, expressive TTS with style modeling and transfer techniques is attracting more attention. In particular, style transfer TTS systems are becoming increasingly important for real-world applications, as they eliminate the need for style-annotated datasets or matched pairs of text and speech data.

Well-designed style encoder is a key component for achieving natural style transfer in TTS systems. The early approaches utilized sentence-level style by applying pooling operations [7, 8, 9]. However, the speaking style cannot be fully expressed by sentence-level style alone, as it has limitations in representing temporal variations in speech, which led to the syllable-level intonation modeling to better capture time-varying style [10]. Furthermore, GenerSpeech [11] explored a more fine-grained approach by extracting the frame-level style. This approach employs multi-level style encoder with vector quantization variational autoencoder [12] to encode style into a discrete codebook, serving as a bottleneck to eliminate non-style information. This approach leads to improvements in style transfer. However, there remains room for further improvement.

Recent works [13, 14] utilized advanced vector quantization methods to improve style extraction such as residual vector quantization (RVQ) and clustering style encoder (CSE). Despite these advances in style extraction methodology, several fundamental challenges remain unsolved. They treat all temporal segments equally, failing to capture the varying importance of different speech regions in extracting speaking style. Additionally, the straight-through estimator [15] used during backpropagation disregards the relative positioning of encoded features within each codebook region, limiting the model's ability to learn fine-grained style details. Finally, relying solely on quantization as a bottleneck without additional constraints does not ensure that extracted style embeddings are independent of content, increasing the risk of content leakage during style transfer. Addressing these issues requires methods that enhance expressiveness while maintaining strong disentanglement from content.

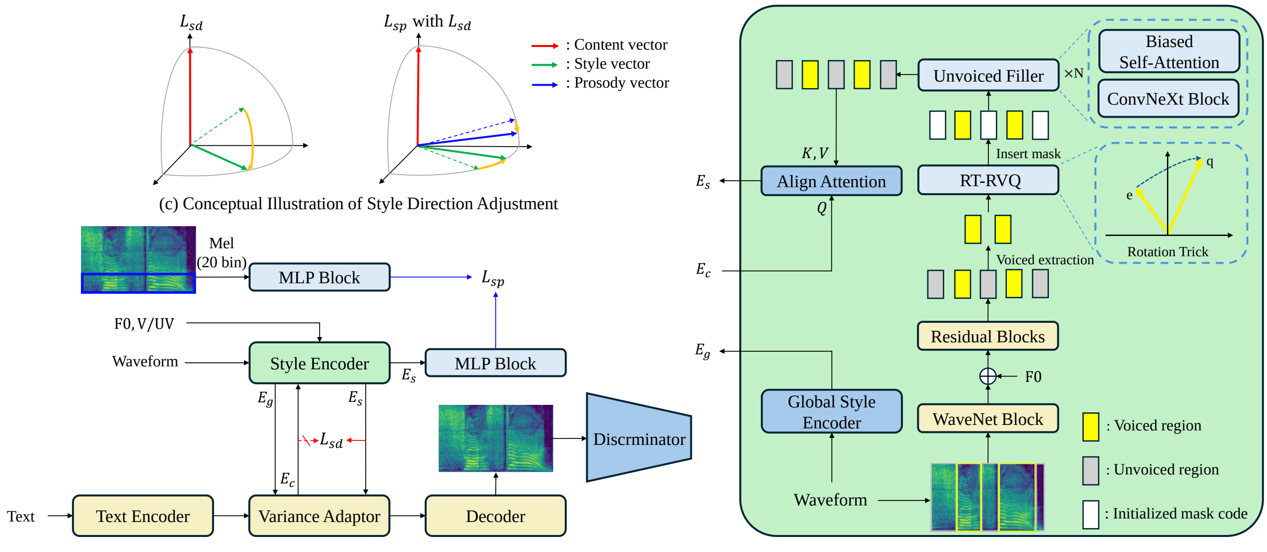
In this paper, we address these limitations through voiced-aware style extraction and style direction adjustment. The importance of region-specific processing has been recognized in signal compression [16], and has recently gained renewed attention in deep learning [17, 18]. Similarly, different regions of speech have varying impacts on speaking style. Voiced regions, which contain rich acoustic information such as harmonics generated by vocal fold vibration, are particularly important for style characteristics. In this context, our voiced-aware style extraction enables effective quantization based on a more compact representation, which enhances the preservation of style-related features with finer granularity. Additionally, we replace the conventional straight-through estimator with a rotation trick [19] in our quantization process, which enables more precise style extraction, particularly in voiced regions. Furthermore, we introduce style direction adjustment, which adjusts the extracted style by modifying its angle using content and prosody vectors in the embedding space. This adjustment process effectively removes content information from style and prevents training instabilities caused by disentanglement. Experimental results demonstrate that the proposed model not only achieves more expressive style transfer but also significantly improves the disentanglement of content and style, leading to more natural speech quality and pronunciation.

2. Spotlight-TTS

In this section, we introduce our proposed model Spotlight-TTS. As shown in Figure 1, we focus on effectively extracting style by considering the importance of different Mel-spectrogram regions while adjusting the direction of style. Our proposed method consists of two parts: voiced-aware style extraction and style direction adjustment.

[†] Corresponding author

[1] https:/Spotlight-TTS.github.io/



(c) Conceptual Illustration of Style Direction Adjustment

(a) Overall Architecture of Spotlight-TTS

(b) Detailed Architecture of Style Encoder

Figure 1: *(a) shows the overall architecture of our proposed Spotlight-TTS. E_c, E_g, and E_s denote the content embedding, global style embedding, and style embedding respectively. (b) shows the details of the style encoder. e and q represent the input feature of the quantization layer and quantized vector respectively. (c) shows the conceptual illustration of angles of vectors changed by style direction adjustment. L_{sd} and L_{sp} represent the style disentanglement loss and style preserving loss.*

2.1. Voiced-aware style extraction

In speech synthesis, we hypothesized that style extraction through codebook learning could be improved by considering different Mel-spectrogram regions that contribute unequally to speaking style. As shown in Figure 1 (b), voiced regions consist of harmonics that are highly correlated with speaking style, and unvoiced regions have simple repetitive patterns that are less related to style. Therefore, we propose voiced-aware style extraction, a novel style extraction method that focuses on the voiced region and fills in the unvoiced region with the unvoiced filler (UF) module.

2.1.1. Voiced frame processing

We use an RVQ module [20] to extract detailed style embeddings. Given that unvoiced regions are less relevant to style compared to voiced regions, we focus RVQ processing on voiced frames through voiced extraction (VE). We use pre-extracted voiced and unvoiced (V/UV) flags to aggregate only voiced frames from the intermediate features as input to the RVQ module. During quantization, we adopt the rotation trick (RT) [19] to improve gradient flow through the quantization layer. Unlike the conventional straight-through estimator, RT preserves the angle between the loss gradient and codebook vector during backpropagation. The RT is computed as:

$$\tilde{q} = sg\left[\frac{||q||}{||e||}R\right]e, \tag{1}$$

where R is the rotation transformation that aligns the input feature e to the closest codebook vector q and $\frac{||q||}{||e||}$ rescales e to match the magnitude of q. Here, $sg\left[\cdot\right]$ denotes the stop-gradient operator used to detach rotation and scaling terms from gradient computation. During forward pass, we use $\tilde{q}$, the transformed input feature, instead of the quantized vector. $\tilde{q}$ is identical to q, in the forward pass, ensuring the RVQ output remains unchanged. By rewriting q in terms of e, the backward pass rotates the gradient to capture the relative position of e in the codebook.

2.1.2. Unvoiced filler module

To improve style continuity between different regions, we propose a UF module. After quantization, we generate learnable mask code embeddings with uniform random initialization and insert them into the unvoiced positions based on their positional information. The UF module then fills these embeddings with meaningful acoustic information. The UF module consists of N identical sub-modules, each of which consists of a ConvNeXt block [21] and a biased self-attention. Biased self-attention enables information flow from the non-masked regions to mask code regions while blocking the opposite direction. Through this asymmetric information flow, the model can utilize the non-masked region without the mask code region negatively affecting the non-masked region. The biased self-attention is defined as follows:

$$Attention(q,k,v) = \left(\text{SoftMax}\left(\frac{qk^T}{\sqrt{d}}\right)\odot\beta\right)v, \tag{2}$$

where β is the attention reweighting (AR) coefficient. Specifically, we define β as 0.02 for mask positions and 1 for non-masked positions to achieve asymmetric information flow.

2.2. Style direction adjustment

After extracting the style, we adjust the directionality of the style to remove content information. To this end, we introduce two complementary losses as conceptually illustrated in Figure 1 (c): style disentanglement (SD) loss encouraging orthogonality between style and content, and style preserving (SP) loss to align the style with prosody.

2.2.1. Style disentanglement loss

Inspired by [22], we use orthogonality loss to disentangle style and content information. Since content information within style embedding can interfere with the learning of content embedding, we detach the content embedding during training. The SD loss L_{sd} is defined as follows:

Table 1: *Comparison with different models for subjective and objective metrics.*

Method	**Style Variation**	**nMOS** (↑)	**sMOS** (↑)	**UTMOS** (↑)	**WER** (↓)	**RMSE**$_{f0}$ (↓)	**RMSE**$_p$ (↓)	**F1 V/UV** (↑)	**SECS** (↑)
GT	-	4.34±0.04	4.63±0.02	3.78	12.42	-	-	-	0.9168
BigVGAN [23]	-	4.30±0.04	4.58±0.03	3.63	12.40	2.45	0.2749	0.8008	0.9150
FastSpeech 2 w/ GST [8]	✗	3.77±0.05	3.05±0.04	3.39	14.18	13.37	0.4619	0.6707	0.8945
FastSpeech 2 w/ CSE [14]	✓	3.74±0.05	3.46±0.04	3.42	13.49	10.39	0.4159	0.7024	0.9013
StyleSpeech [9]	✗	3.74±0.05	3.14±0.04	3.37	13.24	13.88	0.4433	0.6716	0.9008
GenerSpeech [11]	✓	3.98±0.04	3.37±0.04	2.92	16.45	11.20	0.4343	0.6709	0.8848
Spotlight-TTS (Proposed)	✓	**4.26±0.04**	**3.84±0.04**	**3.56**	**12.64**	**8.27**	**0.4050**	**0.7053**	**0.9061**

$$L_{\text{sd}} = \left\| sg\left[E_c\right] E_s^T \right\|_F^2, \tag{3}$$

where E_c and E_s denote the content embedding, style embedding respectively. The expression $\|\cdot\|_F^2$ is the Frobenius norm.

2.2.2. *Style preserving loss*

To enhance stability in style learning and mitigate errors from F0 and V/UV predictions, we introduce an SP loss. This loss refines extracted style embeddings by increasing similarity with low-frequency prosody embeddings. Two separate MLP blocks are used to extract prosody embeddings from the lower 20 bins of the Mel-spectrogram and style embedding, respectively. By increasing the cosine similarity between these two embeddings, we refine the prosody information in the style embedding. The SP loss L_{sp} is defined as follows:

$$L_{\text{sp}} = -\sum_{i=1}^{t} \text{cos\_sim}\big(p_i, \tilde{s}_i\big), \tag{4}$$

where t is a number of timestep. p_i represents the prosody vector extracted from the low-band Mel-spectrogram, and $\tilde{s}_i$ denotes the projected style vector.

2.3. Training stage of Spotlight-TTS

As described in Figure 1 (a), the proposed Spotlight-TTS model consists of a text encoder, a variance adaptor, a decoder, discriminators [24], a style encoder, and MLP blocks. The MLP-blocks and discriminator are employed exclusively during the training phase. For synthesizing the Mel-spectrogram, it is trained using the objective functions L_{fs2} of FastSpeech 2 [1]. L_{rvq} is used to train the RVQ module. As a result, the total loss of the proposed model is formulated as follows:

$$L_{\text{total}} = L_{\text{fs2}} + \lambda_{\text{rvq}} L_{\text{rvq}} + \lambda_{\text{adv}} L_{\text{adv}} + \lambda_{\text{sd}} L_{\text{sd}} + \lambda_{\text{sp}} L_{\text{sp}}, \tag{5}$$

where λ_{rvq}, λ_{adv}, λ_{sd}, and λ_{sp} denotes the weights for each loss term, which are set to 1.0, 0.05, 0.02, and 0.02, respectively.

3. Experiments and results

3.1. Experimental setup

We use the emotional speech dataset (ESD) [25] to verify whether the models can capture style using expressive reference speech. It contains ten English speakers, each producing 350 sentences in five emotions (happy, sad, neutral, surprise, and angry). We follow the original partitioning criteria of the dataset, totaling 17,500 samples. Mel-spectrogram with 80 bins was extracted by short-time Fourier transform with a hop size of 256, a window size of 1,024, and an FFT size of 1,024. We employ the AdamW optimizer [26], setting the hyperparameters β_1 to 0.9 and β_2 to 0.98. Our model was trained for 200k steps on a single NVIDIA RTX 2080Ti GPU. For the audio synthesis in our experiments, we trained the vocoder using the official BigVGAN implementation [23]. We compare our Spotlight-TTS with other FastSpeech2-based style transfer TTS models: FastSpeech2-GST [1, 8], FastSpeech2-CSE [1, 14], StyleSpeech [9], GenerSpeech [11]. For models utilizing time-variant style, we use the same global style encoder [11] for sentence-level style.

3.2. Implementation details

Our TTS system is based on FastSpeech2 [1] and incorporates a multi-length discriminator [24] for improved speech quality. For SP loss, both style embedding and lower 20 bins of Mel-spectrogram are projected to 32 dimensions through MLP blocks from their initial dimensions of 256 and 20, respectively. Each MLP block consists of two linear layers with a GELU activation function [27]. As shown in Figure 1 (b), The style encoder is composed of two parts: a pre-trained global style encoder[2] that extracts the sentence-level style and a voiced-aware style encoder that focuses on frame-level style. The latter consists of a WaveNet block, four convolutional residual blocks, an RVQ module [20], and three UF blocks. The RVQ module has a depth of four and applies RT computed by a householder reflection matrix [19]. After the unvoiced filler blocks, we use scaled dot-product attention [28] to align the time dimension of the style embedding with that of the content embedding. The aligned style embedding is utilized within the variance adaptor and added to its final output along with the global style embedding.

3.3. Evaluation metrics

To evaluate the speech quality, we conduct both objective and subjective evaluations of the synthesized speech. For the subjective evaluation, we use the naturalness mean opinion score (MOS) and similarity mean opinion score (sMOS). Both metrics are rated from 1 to 5 with a confidence interval of 95%. We randomly select 50 samples from the test set, with 5 samples per speaker. Each audio has been listened to by 20 participants. Additionally, we utilize the open-source UTMOS[3] [29] as a MOS prediction model for the naturalness metric. To evaluate linguistic consistency, we calculate the word error rate (WER) by Whisper[4] [30]. For the speaker similarity measurements, we calculate the speaker embedding cosine similarity (SECS) via WavLM[5] [31]. For prosodic evaluation, we compute the root mean square error for both pitch error (RMSE$_{f_0}$) measured in Hz and periodicity error (RMSE$_p$), along with the F1 score of voiced/unvoiced classification (F1$_{v/uv}$). All objective metrics are computed using the entire test set. For reference audio selection, we used same strategy as in [11].

[2] https://github.com/Rongjiehuang/GenerSpeech/tree/encoder

[3] https://github.com/tarepan/SpeechMOS

[4] https://github.com/openai/whisper

[5] https://huggingface.co/microsoft/wavlm-base-sv

Table 2: *Results of AXY preference test on parallel and non-parallel style transfer.*

Baseline	**Setting**	**7-point score**	**Preference (%)**		
			X	Neutral	Y
FastSpeech2 w/ GST	Parallel	1.21 ± 0.12	15%	22%	63%
	Non-Parallel	0.59 ± 0.11	28%	16%	56%
FastSpeech2 w/ CSE	Parallel	0.53 ± 0.12	28%	17%	55%
	Non-Parallel	0.33 ± 0.15	29%	29%	42%
StyleSpeech	Parallel	1.16 ± 0.11	17%	21%	62%
	Non-Parallel	0.25 ± 0.12	34%	22%	44%
GenerSpeech	Parallel	0.95 ± 0.12	17%	36%	47%
	Non-Parallel	0.38 ± 0.13	24%	33%	43%

3.4. Model performance

As shown in Table 1, our proposed Spotlight-TTS achieves significant improvements in both subjective and objective metrics. Global style-based models like FastSpeech2 w/GST and StyleSpeech demonstrate lower performance in style-related metrics due to the loss of temporal style variations through pooling operations. In contrast, FastSpeech2 w/CSE and GenerSpeech outperform global style-based models across all style-related metrics by preserving temporal dynamics. Furthermore, our model surpasses baselines across all metrics by considering temporal dynamics with region-specific importance and directionality of extracted style.

We also evaluate style transfer performance through an AXY test [11] with scores ranging from -3 to 3, where 0 denotes "Both are about the same distance". X, Neutral, Y denote baseline, same and ours respectively. As shown in Table 2, our subjective evaluation demonstrates superior style transfer performance across both parallel and non-parallel settings compared to baseline models. Similar to Table 1, time-variant style-based models outperform global style-based models. Spotlight-TTS further advances this approach by explicitly modeling the relative importance of different temporal regions based on their acoustic characteristics. Unlike previous time-variant style-based models that treated all temporal regions equally, our region-aware style extraction enables more effective style capture, leading to superior style transfer performance. Additionally, our style direction adjustment disentangles content information from style in the embedding space, enabling robust style extraction even when linguistic content differs between reference speech and input text.

3.5. Ablation study

3.5.1. Voiced-aware style extraction

We evaluated voice-aware style extraction by removing key components and analyzing their effects. As shown in Table 3, removing RT leads to degraded performance, particularly in style-related metrics. This indicates that the rotation trick allows the RVQ to better capture the harmonic structures in voiced regions by ensuring stable gradient propagation. When both the RT and the UF are removed, we observe further performance degradation. Although unvoiced regions have relatively simple and repetitive patterns compared to voiced regions, removing the UF module leads to degraded pronunciation and prosody. This indicates that while unvoiced regions are less critical for style, proper handling of these regions through the UF module is still necessary. Additionally, removing VE results in significantly increased pitch error. This demonstrates that our voiced-aware approach, which aggregates voiced frames to focus quantization on style-rich regions, is essential for capturing detailed style information.

Table 3: *Results of ablation study on voiced-aware style extraction and style direction adjustment.*

Method	**nMOS**	**WER**	**RMSE$_{f0}$**	**RMSE$_p$**	**F1 V/UV**
Ours	**3.93±0.07**	**12.64**	**8.27**	0.4050	**0.7053**
−RT	3.91±0.06	13.24	9.43	0.4154	0.6928
−RT −UF	3.91±0.07	13.41	9.82	0.4274	0.6874
−RT −UF −VE	3.84±0.07	14.06	11.48	0.4425	0.6829
−SP	3.86±0.06	13.66	9.74	0.4297	0.6915
−SP −SD	3.66±0.07	15.38	8.53	**0.4037**	0.6848

Table 4: *Results of ablation study on biased self-attention in unvoiced filler.*

Method	**nMOS**	**WER**	**RMSE$_{f0}$**	**RMSE$_p$**	**F1 V/UV**
Ours	**3.95±0.07**	**12.64**	**8.27**	**0.4050**	**0.7053**
Self-attention w/ BM	3.93±0.07	13.61	13.19	0.4528	0.6751
Self-attention	3.87±0.07	13.64	16.38	0.4587	0.6668

3.5.2. Style direction adjustment

We also investigate the effectiveness of our style direction adjustment mechanism. When the SP loss is removed, we observe highly increased pitch errors. This suggests that SP loss effectively mitigates the degradation of prosodic information caused by the strong constraints of SD loss. Removing both SP and SD losses results in severely degraded nMOS and WER indicating that removing content information from style is important for both speech quality and pronunciation.

3.5.3. Biased self-attention

To investigate the effectiveness of biased self-attention, we conduct an additional ablation in Table 4. If we replace AR with a simple binary mask (BM) with 1 and 0, the model completely blocks information flow from non-masked to mask code regions, preventing the mask code embeddings from being filled with meaningful acoustic information. This disrupts the natural prosodic continuity between non-masked and mask code regions, resulting in higher F0 errors and worse V/UV classification. In contrast, using conventional self-attention allows excessive interference between regions, significantly degrading both metrics. These results validate that our biased self-attention enables optimal information flow between voiced and unvoiced regions.

4. Conclusion

We presented Spotlight-TTS, a framework for synthesizing expressive speech by focusing on voiced regions in the Mel-spectrogram and adjusting the direction of the extracted style. Voiced-aware style extraction considers the acoustic characteristics of different speech regions, enabling more detailed style extraction. Furthermore, the style direction adjustment effectively disentangles content from style, while preserving prosody information within the style embedding. Experimental results demonstrate that our method generates more natural, expressive speech while achieving these improvements through style-focused modifications. Despite these advances, there remains room for improvement in non-parallel style transfer, especially when reference speech duration significantly differs from the input text. Future work will focus on improving speech quality in non-parallel settings. Nevertheless, our findings highlight the effectiveness of sophisticated style modeling, offering a promising direction for expressive TTS systems.

5. Acknowledgements

This work was partly supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (Artificial Intelligence Graduate School Program (Korea University) (No. RS-2019-II190079), Artificial Intelligence Innovation Hub (No. RS-2021-II212068), AI Technology for Interactive Communication of Language Impaired Individuals (No. RS-2024-00336673), and Artificial Intelligence Star Fellowship Support Program to Nurture the Best Talents (IITP-2025-RS-2025-02304828)).

6. References

[1] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech 2: Fast and high-quality end-to-end text to speech," in *International Conference on Learning Representations*, 2021.

[2] S.-W. Lee and H.-H. Song, "A new recurrent neural-network architecture for visual pattern recognition," *IEEE Transactions on Neural Networks*, vol. 8, no. 2, pp. 331–340, 1997.

[3] S.-W. Lee and Y.-J. Kim, "Multiresolution recognition of handwritten numerals with wavelet transform and multilayer cluster neural network," in *Proceedings of 3rd International Conference on Document Analysis and Recognition*, vol. 2, 1995, pp. 1010–1013 vol.2.

[4] S.-W. Lee, "Multilayer cluster neural network for totally unconstrained handwritten numeral recognition," *Neural Networks*, vol. 8, no. 5, pp. 783–792, 1995.

[5] S.-H. Lee, S.-B. Kim, J.-H. Lee, E. Song, M.-J. Hwang, and S.-W. Lee, "Hierspeech: Bridging the gap between text and speech by hierarchical variational inference using self-supervised representations for speech synthesis," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 16 624–16 636.

[6] D.-H. Cho, H.-S. Oh, S.-B. Kim, S.-H. Lee, and S.-W. Lee, "Emosphere-tts: Emotional style and intensity modeling via spherical emotion vector for controllable emotional text-to-speech," in *Interspeech 2024*, 2024, pp. 1810–1814.

[7] R. Skerry-Ryan, E. Battenberg, Y. Xiao, Y. Wang, D. Stanton, J. Shor, R. Weiss, R. Clark, and R. A. Saurous, "Towards end-to-end prosody transfer for expressive speech synthesis with tacotron," in *International Conference on Machine Learning*, vol. 80, 2018, pp. 4693–4702.

[8] Y. Wang, D. Stanton, Y. Zhang, R.-S. Ryan, E. Battenberg, J. Shor, Y. Xiao, Y. Jia, F. Ren, and R. A. Saurous, "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis," in *International Conference on Machine Learning*, vol. 80, 2018, pp. 5180–5189.

[9] D. Min, D. B. Lee, E. Yang, and S. J. Hwang, "Meta-stylespeech : Multi-speaker adaptive text-to-speech generation," in *International Conference on Machine Learning*, vol. 139, 2021, pp. 7748–7759.

[10] H. Tang, X. Zhang, J. Wang, N. Cheng, and J. Xiao, "Qi-tts: Questioning intonation control for emotional speech synthesis," in *2023 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2023, pp. 1–5.

[11] R. Huang, Y. Ren, J. Liu, C. Cui, and Z. Zhao, "Generspeech: Towards style transfer for generalizable out-of-domain text-to-speech," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 10 970–10 983.

[12] A. van den Oord, O. Vinyals, and k. kavukcuoglu, "Neural discrete representation learning," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[13] D. Seong, H. Lee, and J.-H. Chang, "Tsp-tts: Text-based style predictor with residual vector quantization for expressive text-to-speech," in *Interspeech 2024*, 2024, pp. 1780–1784.

[14] Y. Zhang, Z. Jiang, R. Li, C. Pan, J. He, R. Huang, C. Wang, and Z. Zhao, "TCSinger: Zero-shot singing voice synthesis with style transfer and multi-level style control," in *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 2024, pp. 1960–1975.

[15] Y. Bengio, N. Léonard, and A. Courville, "Estimating or propagating gradients through stochastic neurons for conditional computation," *arXiv preprint arXiv:1308.3432*, 2013.

[16] N. Jayant, J. Johnston, and R. S. Safranek, "Signal compression based on models of human perception," *Proceedings of the IEEE*, vol. 81, no. 10, pp. 1385–1422, 1993.

[17] M. Huang, Z. Mao, Q. Wang, and Y. Zhang, "Not all image regions matter: Masked vector quantization for autoregressive image generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 2002–2011.

[18] Z. Liu, J. Gui, and H. Luo, "Good helper is around you: Attention-driven masked image modeling," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 2, pp. 1799–1807, 2023.

[19] C. Fifty, R. G. Junkins, D. Duan, A. Iyengar, J. W. Liu, E. Amid, S. Thrun, and C. Re, "Restructuring vector quantization with the rotation trick," in *International Conference on Learning Representations*, 2025.

[20] D. Lee, C. Kim, S. Kim, M. Cho, and W.-S. Han, "Autoregressive image generation using residual quantization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 523–11 532.

[21] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020s," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 976–11 986.

[22] 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 Trans. Audio, Speech and Lang. Proc.*, vol. 30, p. 1448–1460, 2022.

[23] S.-g. Lee, W. Ping, B. Ginsburg, B. Catanzaro, and S. Yoon, "Bigvgan: A universal neural vocoder with large-scale training," in *International Conference on Learning Representations*, 2023.

[24] Z. Ye, Z. Zhao, Y. Ren, and F. Wu, "Syntaspeech: Syntax-aware generative adversarial text-to-speech," in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 2022, pp. 4468–4474, main Track.

[25] K. Zhou, B. Sisman, R. Liu, and H. Li, "Emotional voice conversion: Theory, databases and esd," *Speech Communication*, vol. 137, pp. 1–18, 2022.

[26] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2019.

[27] D. Hendrycks and K. Gimpel, "Gaussian error linear units (gelus)," *arXiv preprint arXiv:1606.08415*, 2016.

[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[29] T. Saeki, D. Xin, W. Nakata, T. Koriyama, S. Takamichi, and H. Saruwatari, "Utmos: Utokyo-sarulab system for voicemos challenge 2022," in *Interspeech 2022*, 2022, pp. 4521–4525.

[30] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. Mcleavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *International Conference on Machine Learning*, vol. 202, 2023, pp. 28 492–28 518.

[31] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao *et al.*, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.