Interspeech 2024
1-5 September 2024, Kos, Greece

ClariTTS: Feature-ratio Normalization and Duration Stabilization for Code-mixed Multi-speaker Speech Synthesis

Changhwan Kim[1]

[1]Hyundai Motor Group, Seoul, Korea
changhwan.kim@hyundai.com

Abstract

Recent text-to-speech (TTS) models have synthesized remarkably natural speech for code-mixed TTS as well as cross-lingual TTS. However, code-mixed texts are synthesized with unnatural accents for each word because speaker-related features can include linguistic features from the speaker's source language. To solve the problems, we propose ClariTTS, which synthesizes speech with appropriate accents for the language of each word in code-mixed texts. Specifically, we propose feature-ratio normalized affine coupling layer in the flow-based TTS model, which disentangles speaker and linguistic features to prevent the accent of the target speaker's source language from being included in the target language. Furthermore, we introduce a duration stabilization training objectives to ensure stable duration prediction in code-mixed TTS. From the experimental results, we demonstrate that ClariTTS reliably generates code-mixed speech with clear pronunciation while preserving speaker identity.

Index Terms: speech synthesis, cross-lingual, code-mixing, Disentangled representation learning

1. Introduction

Over the years, end-to-end neural network-based text-to-speech (TTS) systems have been able to synthesize human-like speech [1, 2, 3, 4, 5, 6]. In addition, research has extended to synthesize speech with multiple speaking styles such as speaker identity and emotion type in a single model [7, 8, 9, 10, 11]. With the increasing need for speech synthesis for multiple languages, many studies have been explored to generate multilingual speech by multiple speakers. However, since collecting data of multilingual speakers is difficult and time-consuming, most approaches implement multi-lingual TTS by combining a monolingual data. Furthermore, when training multi-lingual TTS model with monolingual data, the accent of the speaker's source language is inevitably contained in the target language in cross-lingual synthesis, possibly resulting in an unnatural accent; i.e., it is a speaker-language entanglement problem.

To overcome this issue, many studies have attempted to mitigate entanglement of speaker identity and linguistic information [12, 13, 14, 15, 16, 17]. For example, domain adversarial training [12, 13, 15] was used to separate speaker identity and linguistic features from each other. Additionally, J. Ye et al. [16] improved cross-lingual pronunciation by constructing triplet training scheme. J.-H. Kim et al. [17] alleviated speaker-language entanglement through dividing acoustic representations into speaker-dependent and speaker-independent.

In addition to the improvement of cross-lingual TTS, recent studies have also considered about code-mixed texts which are two or more languages included in one sentence. For example, code-mixed TTS models have generated adapting the encoder structure to suit the code-mixed TTS [18], utilizing additional features derived from pre-trained models from other domain [19, 20], or enriching text input by employing transliteration [21, 22]. However, the above approaches have limitations that they are influenced by the performance of pre-trained external models or transliteration.

Therefore, we propose ClariTTS, which improves naturalness of code-mixed speech synthesis by feature-ratio normalization (FRN), denormalization ($FRDN$) and duration stabilization training objectives. Specifically, we utilize normalization-based conditioning method [23] in the affine coupling layers of flow-based TTS model [5]. In the training phase, we use separately predicted parameters for the speaker and language embeddings to normalize the inputs respectively and add these speaker and language-normalized results. At this time, we compute the ratio, which adaptively determines the proportion of adding speaker and language-normalized results. As a result, the normalizing flow converts speaker and language-dependent data distribution to speaker and language-independent latent prior distribution. Since we separately used speaker and language normalization, the proposed model explicitly disentangles speaker and language features at training. This enables that the affine coupling layer injects appropriate speaker and language information through denormalization at inference.

Moreover, we design duration stabilization training objectives to further mitigate speaker-language entanglement as well as help the duration predictor to be robust. We divide duration loss into intra-speaker and cross-speaker duration loss. First, we predict intra-speaker duration using input text, language embeddings and speaker embeddings that are paired in the mini-batch. Meanwhile, we randomly shuffle speaker embeddings along the batch dimension and predicts cross-speaker duration by the shuffled speaker embeddings. Since the shuffled speaker embeddings contain different speaker's identity and/or language information to the paired speaker embeddings of mini-batch, the duration predictor increasingly obtains language information from the input text and language embeddings. Experimental results indicate that ClariTTS generates intra/cross-lingual and code-mixed speech with clear pronunciation while preserving speaker identity. Furthermore, we demonstrate that our proposed model effectively determine the proportion of speaker and language information for each channel through the affine coupling layer. Our contributions are summarized as below:

- We propose a flow-based TTS model that adaptively learns the utilization ratio of speaker and language features as well as explicitly disentangles speaker and language features.
- We propose duration stabilization training objectives to further alleviate speaker-language entanglement and reliably train the duration predictor.

10.21437/Interspeech.2024-608

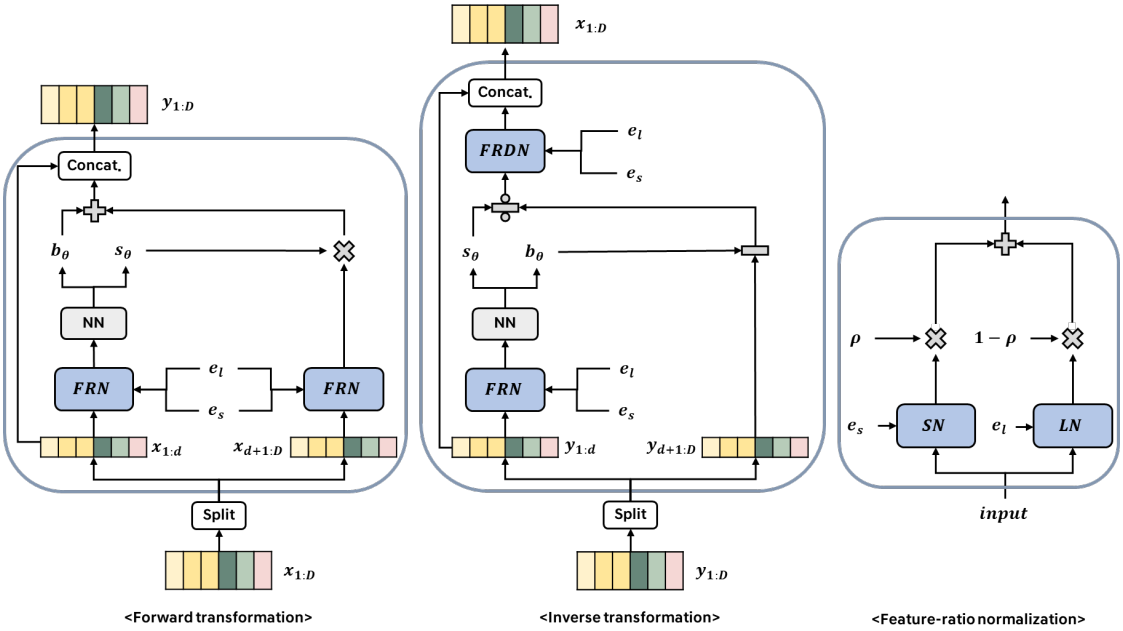

Figure 1: *Forward (left), inverse transformation (center) and proposed normalization (right) of the affine coupling layers of ClariTTS.*

2. Related work

In the flow-based TTS models [3, 5], there are normalizing flows that learn bijective mapping between simple prior distribution and complex data distribution through efficient invertible transformation such as affine coupling layer. At this time, B. J. Choi et al. [23] introduced speaker-normalized affine coupling layer (SNAC). They explicitly normalized the input by the predicted mean and standard deviation parameters of speaker embedding (e_s) at training while denormalizing input by the desired speaker embedding at inference. Specifically, first, the normalization g and denormalization g^{-1} are defined as

$$g(x;c) = \frac{x - m_\theta(c)}{\exp(v_\theta(c))} \tag{1}$$

$$g^{-1}(x;c) = x \odot \exp(v_\theta(c)) + m_\theta(c) \tag{2}$$

where $x \in \mathbb{R}^D$, c and $\odot$ are input, condition and element-wise product, and m_θ, v_θ are the simple linear projections to obtain mean and standard deviation parameters from c. From the above definitions, speaker normalization (SN) and speaker denormalization (SDN) can be obtained, e.g., $SN(x;e_s) = g(x;e_s)$ and $SDN(x;e_s) = g^{-1}(x;e_s)$. Subsequently, the forward transformation of SNAC layer can be obtained by utilizing SN to the affine coupling layer.

$$y_{1:d} = x_{1:d} \tag{3}$$

$$y_{d+1:D} = SN(x_{d+1:D};e_s) \odot \exp(s_\theta(SN(x_{1:d};e_s))) + b_\theta(SN(x_{1:d};e_s)) \tag{4}$$

Also, the inverse transformation is performed as

$$x_{1:d} = y_{1:d} \tag{5}$$

$$x_{d+1:D} = SDN\left(\frac{y_{d+1:D} - b_\theta(SN(y_{1:d};e_s))}{\exp(s_\theta(SN(y_{1:d};e_s)))};e_s\right) \tag{6}$$

It can be interpreted as removing speaker information during the forward process and providing it during the inverse process, respectively. Therefore, they converted speaker-dependent data distribution into the speaker-independent prior distribution, enabling the model to obtain the desired speaker-dependent data distribution through the inverse process. We extend the above method to speaker-language conditioning method that add normalized inputs by each of the speaker and language embeddings at training and denormalized inputs at inference. We replace SN and SDN to FRN and $FRDN$, which selectively normalize and denormalize input by the information of speaker and language embeddings.

Meanwhile, in computer vision, researchers have attempted to selectively refine or manipulate the hidden features by utilizing attention-based feature refinement to improve representation power [24] and combining different normalization techniques to selectively normalize style encoded by each feature map based on learnable ratio [25]. Inspired by the above works, we compute the feature-ratio, which determines the proportion of normalization for the speaker and language embeddings for each channel of input in the affine coupling layer.

In the cross-lingual TTS, several studies have utilized the regularization loss to mitigate speaker-language entanglement and generate cross-lingual speech with appropriate duration. SANE-TTS [15] introduced a speaker regularization loss that helps the duration predictor reliably predict the duration in cross-lingual synthesis by pushing mean of speaker representations to zero vector. As a result, at the inference stage, the duration predictor receives a zero vector and generate moderate duration in cross-lingual synthesis. Crossspeech [17] improved generalization performance by reducing gaps between text encodings with mixed [26] and original speaker representations. From these perspectives, we propose duration stabilization loss that causes the cross-speaker duration predicted by the shuffled speaker embeddings to follow the intra-speaker duration predicted by the original speaker embddings. As a result, the duration predictor does not leverage the language information of speaker embeddings, but only utilizes speaker-related features to generate duration.

3. Method

We used VITS [5] as a backbone of ClariTTS, which includes a text encoder, duration predictor, normalizing flow, posterior encoder and decoder. For each language, we used native characters and one-hot language ID [14]. We also utilized the reference encoder [7] to extract speaker-related features from the linear-scale spectrogram. We used the outputs of the reference encoder as speaker embeddings. For fast inference speed, we replaced the decoder parts of VITS with those of multi-stream inverse short-time Fourier transform VITS (MS-iSTFT-VITS) [6]. Moreover, we used deterministic duration predictor instead of stochastic duration predictor for stable inference [6, 14, 15]. Based on the above TTS system, we applied two proposed methods: 1) FRN, $FRDN$ in the affine coupling layers of the normalizing flow and 2) duration stabilization training objectives for the duration predictor. Since our main objective is to disentangle speaker and language features, we provide speaker and language embeddings only to the normalizing flow and duration predictor rather than other modules. Further details are explained in 4.1, and we will describe the details of our proposed methods in the following subsections.

3.1. Feature-ratio normalized affine coupling layer

Fig.1 depicts the forward and inverse transformation of our proposed affine coupling layer in the normalizing flow. The architecture is identical to the SNAC layer [23], but we replaced SN and SDN to FRN and $FRDN$. The blue boxed parts of the architecture represent FRN and $FRDN$, which are normalized and denormalized by mean and standard deviation parameters obtained from speaker embeddings (e_s) and language embeddings (e_l), respectively. First, substituting c into e_l from equations (1) and (2) yields language normalization (LN) and language denormalization (LDN). Simultaneously, we compute the feature-ratio ρ using e_s and e_l from the shared convolutional neural networks W_r [24] as given by

$$\rho = \sigma(W_r(m_\theta(e_s), v_\theta(e_s)) + W_r(m_\theta(e_l), v_\theta(e_l))) \quad (7)$$

where σ denotes sigmoid function. Thus, we represent FRN as follows:

$$FRN(x; e_{s,l}) = \rho(SN(x; e_s)) + (1-\rho)(LN(x; e_l)) \quad (8)$$

and $FRDN$ is inverse transformation of FRN. Finally, we can derive forward transformation of the affine coupling layer as

$$\begin{aligned} y_{1:d} &= x_{1:d} \\ y_{d+1:D} &= FRN(x_{d+1:D}; e_{s,l}) \odot \exp(s_\theta(FRN(x_{1:d}; e_{s,l}))) \\ &\quad + b_\theta(FRN(x_{1:d}; e_{s,l})) \end{aligned} \quad (9)$$

where s_θ and b_θ are the scale and bias functions and $d < D$. Also, the inverse transformation is represented as follows:

$$\begin{aligned} x_{1:d} &= y_{1:d} \\ x_{d+1:D} &= FRDN\left(\frac{y_{d+1:D} - b_\theta(FRN(y_{1:d}; e_{s,l}))}{\exp(s_\theta(FRN(y_{1:d}; e_{s,l})))}; e_{s,l}\right) \end{aligned} \quad (10)$$

Because of coupling structure, the Jacobian becomes a lower triangular matrix [23]. Thus, the Jacobian can be derived by

$$\begin{aligned} \frac{\partial y_{d+1:D}}{\partial x_{d+1:D}} =& \text{diag}\Bigg(\exp(s_\theta(FRN(x_{1:d}; e_{s,l}))) \\ & \odot \frac{\rho \exp(v_\theta(e_l)) + (1-\rho)\exp(v_\theta(e_s))}{\exp(v_\theta(e_s))\exp(v_\theta(e_l))} \Bigg) \end{aligned} \quad (11)$$

Table 1: *The number of model training parameters of ClariTTS and baselines.*

Method	#Params.
MS-iSTFT-VITS	43.83M.
YourTTS	43.83M.
SANE-TTS	44.37M.
ClariTTS	40.49M.

$$\frac{\partial y}{\partial x} = \begin{bmatrix} I_{d\times d} & 0 \\ \frac{\partial y_{d+1:D}}{\partial x_{1:d}} & \frac{\partial y_{d+1:D}}{\partial x_{d+1:D}} \end{bmatrix} \quad (12)$$

where $I_{d\times d}$ denotes a $d\times d$ identity matrix. As in the case of [5], for simplicity, we designed the normalizing flow to be the volume-preserving transformation. It means that the output of scale function $\exp(s_\theta(FRN(x_{1:d}; e_{s,l})))$ becomes one. Therefore, the log-determinant of Jacobian of the proposed normalizing flow can be derived by

$$\begin{aligned} &\log\left|\det \frac{\partial f_\theta(x)}{\partial x}\right| \\ &= \log \sum_j \frac{\rho \exp(v_\theta(e_l)_j) + (1-\rho)\exp(v_\theta(e_s)_j)}{\exp(v_\theta(e_s)_j)\exp(v_\theta(e_l)_j)} \end{aligned} \quad (13)$$

In summary, via FRN, the affine coupling layer removes speaker and language information in the input in the forward transformation. At this stage, the proposed layer adaptively eliminates speaker and language information using ρ for each hidden channel, allowing the normalizing flow to convert the data distribution into the speaker and language-independent latent prior distribution. On the contrary, during the inverse transformation, $FRDN$ provides information through the embeddings of the target speaker and language, resulting in the conversion of the prior distribution into the speaker and language-dependent data distribution.

3.2. Duration stabilization training objectives

In Section 1, we argued that the problem of cross-lingual TTS is due to the speaker-language entanglement. We address this by FRN and FRDN, and we propose training objectives that stabilizes the duration predictor and further mitigates speaker-language entanglement. Specifically, given the mini-batch, there are paired input data such as $(text, audio, e_s, e_l)$, respectively. The duration predictor generates intra-speaker duration d_{intra} as given by

$$d_{intra} = W_d(x; e_{s,l}) \quad (14)$$

where W_d is duration predictor and x is input text embeddings. Meanwhile, we obtain shuffled speaker embeddings $\widetilde{e_s} = \text{shuffle}(e_s)$ by randomly shuffle e_s in the mini-batch. If the duration predictor generates duration with $\widetilde{e_s}$, it can be interpreted that the duration predictor generates cross-speaker duration d_{cross} because $\widetilde{e_s}$ contains information from other speakers and/or languages compared to e_s. Thus, we define duration stabilization loss as follows:

$$\begin{aligned} \mathcal{L}_{dur} &= \mathcal{L}_{d_{intra}} + \mathcal{L}_{d_{cross}} \\ &= \text{MSE}(d_{mas}, d_{intra}) + \text{MSE}(d_{mas}, d_{cross}) \end{aligned} \quad (15)$$

where MSE and d_{mas} refer to mean square error loss and duration by monotonic alignment search [3]. Through $\mathcal{L}_{dur}$, the

Table 2: *MOS, SMOS, WER and CER evaluation results. MOS and SMOS are represented with 95% confidence interval.*

Method	Intra-lingual				Cross-lingual				Code-mixed	
	MOS	SMOS	WER	CER	MOS	SMOS	WER	CER	MOS	SMOS
Ground truth	4.70 ± 0.12	4.59 ± 0.12	7.29	2.1	-	-	-	-	-	-
MS-iSTFT-VITS	3.93 ± 0.14	3.61 ± 0.2	11.90	4.14	3.09 ± 0.16	3.29 ± 0.19	17.68	7.54	3.34 ± 0.16	3.18 ± 0.19
YourTTS	3.85 ± 0.17	3.95 ± 0.17	10.46	3.40	3.38 ± 0.17	3.62 ± 0.14	14.25	5.89	3.34 ± 0.17	3.55 ± 0.17
SANE-TTS	3.76 ± 0.17	3.93 ± 0.18	11.00	5.13	3.35 ± 0.17	3.68 ± 0.16	13.92	6.49	3.28 ± 0.16	3.79 ± 0.13
ClariTTS	$\mathbf{4.36 \pm 0.11}$	$\mathbf{4.29 \pm 0.15}$	**8.17**	**2.58**	$\mathbf{3.90 \pm 0.14}$	$\mathbf{3.83 \pm 0.15}$	**9.54**	**3.73**	$\mathbf{3.91 \pm 0.11}$	$\mathbf{4.21 \pm 0.13}$
w/o $\mathcal{L}_{d_{cross}}$	4.23 ± 0.10	4.03 ± 0.07	9.81	3.16	3.79 ± 0.14	3.80 ± 0.13	10.90	3.96	3.78 ± 0.12	4.08 ± 0.09
w/o $FRN, \mathcal{L}_{d_{cross}}$	4.16 ± 0.12	4.01 ± 0.18	9.22	2.65	3.59 ± 0.17	3.67 ± 0.2	11.31	4.28	3.29 ± 0.17	4.03 ± 0.15

duration predictor extracts speaker-related features without extracting linguistic features from speaker embeddings. Therefore, the duration predictor is robust against cross-lingual TTS because it utilizes the role of speaker and language embedding separately.

4. Experiment

4.1. Experimental setting

Dataset and preprocessing We used two monolingual datasets to train our model. For Korean, we used subset of AIHub [27] multi-speaker dataset, which contain 2000 hours of speech from 950 speakers. For English, we used LibriTTS-R [28], a restored version of LibriTTS containing 555 hours of speech by 2,311 speakers. Native characters were used for each language, and each utterance was downsampled to 22050 Hz. Preprocessing of English text followed the method of VITS [5], and for Korean, characters unrelated to Hangeul were removed or converted into Hangeul.

Model and training details We used transformer-based text encoder consisting of 10 layers of transformer block. To generate ratio ρ, we used two 1d CNN layers with input and output channel sizes of $(192, 6)$ and $(6, 96)$, respectively. We use 4-dimensional language embeddings [14]. We have discussed other details in section 3, while the remaining aspects not covered here are consistent with VITS.

We compared ClariTTS with three other TTS models: MS-iSTFT-VITS [6], YourTTS [14] and SANE-TTS [15]. For pair comparison, decoder parts of YourTTS and SANE-TTS were replaced with those of MS-iSTFT-VITS. Moreover, we also used 10 layers of transformer block for text encoder and utilized deterministic duration predictor of baselines. For baseline models, we input speaker and language embeddings to the duration predictor, normalizing flow, posterior encoder and decoder. We trained ClariTTS and three baseline models using two A100 GPUs, each trained on batch size 64 and 800k steps for about a week per model.

Evaluation metrics To evaluate the performance of ClariTTS, we utilized several subjective and objective metrics. We conducted mean opinion score (MOS) tests by asking 19 bilingual listeners to score the quality of audio on a 1 to 5 scale in terms of naturalness and intelligibility. Likewise, speaker similarity MOS (SMOS) test was conducted to measure speaker similarity. For the above subjective tests, 30 samples were used for each model. Moreover, we used Whisper [29] to measure word error rate (WER) and character error rate (CER) to evaluate the clarity of the synthesized speech as objective metrics. We synthesized 100 sentences created by ChatGPT[30], and measured WER and CER for a total of 500 sentences. Code-mixed samples were synthesized by manually generated code-mixed sentences.

4.2. Experimental results

Table 1 and 2 summarizes the number of parameters and evaluation results of the baselines and ClariTTS. The MOS and SMOS tests show that ClariTTS outperforms the baseline methods in terms of naturalness, intelligibility and speaker similarity with less number of parameters. While our proposed method aims to improve the overall quality of code-mixed speech, it is worth noting that intra-lingual and cross-lingual speeches also have improved synthesis quality. Furthermore, experimental results indicate that ClariTTS synthesizes more clear speech objectively proved by WER and CER. Audio samples are available at our demo page[1].

4.3. Ablation study

To investigate effectiveness of our proposed methods, we conducted an ablation study by eliminating each proposed method. As the result at the bottom of Table 2, both cross-speaker duration loss and FRN improves synthesis quality. Specifically, when cross-speaker duration loss is not used, duration predictor does not see cross-lingual cases in training. Therefore, the duration predictor become less stable for for cross-lingual or code-mixed speech, resulting in quality degradation. In addition, when FRN and $FRDN$ are not used, the TTS model does not disentangle speaker and language features. This means that the speaker and language features are entangled; the cross-lingual and code-mixed synthesis quality are further degraded. In summary, FRN and cross-lingual duration loss are both effective in improving cross-lingual and code-mixed speech synthesis.

5. Conclusion

We propose ClariTTS which synthesizes cross-lingual and code-mixed speech with high naturalness, clarity and speaker similarity. To mitigate speaker-language entanglement, we proposed FRN, FRDN and cross-speaker duration loss. FRN explicitly disentangle speaker and language information by separately utilize speaker and language normalization based on feature-ratio ρ. Also, FRDN adaptively inject speaker and language normalization based on ratio in the inference stage. Moreover, cross-speaker duration loss further mitigate speaker-language entanglement with stable duration generation. Our experimental results proved that ClariTTS outperforms baseline methods in terms of naturalness, intelligibility and speaker similarity. Although our experiments aim to explore cross-lingual and code-mixed speech, we believe that our research can be extended to encompass multiple languages.

[1] https://claritts.github.io/

6. References

[1] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan *et al.*, "Natural tts synthesis by conditioning wavenet on mel spectrogram predictions," in *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2018, pp. 4779–4783.

[2] 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*, 2020.

[3] J. Kim, S. Kim, J. Kong, and S. Yoon, "Glow-tts: A generative flow for text-to-speech via monotonic alignment search," *Advances in Neural Information Processing Systems*, vol. 33, pp. 8067–8077, 2020.

[4] A. Łańcucki, "Fastpitch: Parallel text-to-speech with pitch prediction," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6588–6592.

[5] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," in *International Conference on Machine Learning*. PMLR, 2021, pp. 5530–5540.

[6] M. Kawamura, Y. Shirahata, R. Yamamoto, and K. Tachibana, "Lightweight and high-fidelity end-to-end text-to-speech with multi-band generation and inverse short-time fourier transform," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[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*. PMLR, 2018, pp. 4693–4702.

[8] Y.-J. Zhang, S. Pan, L. He, and Z.-H. Ling, "Learning latent representations for style control and transfer in end-to-end speech synthesis," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6945–6949.

[9] R. Valle, J. Li, R. Prenger, and B. Catanzaro, "Mellotron: Multispeaker expressive voice synthesis by conditioning on rhythm, pitch and global style tokens," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6189–6193.

[10] R. Terashima, R. Yamamoto, E. Song, Y. Shirahata, H.-W. Yoon, J.-M. Kim, and K. Tachibana, "Cross-Speaker Emotion Transfer for Low-Resource Text-to-Speech Using Non-Parallel Voice Conversion with Pitch-Shift Data Augmentation," in *Proc. Interspeech 2022*, 2022, pp. 3018–3022.

[11] C. Kim, S.-y. Um, H. Yoon, and H.-G. Kang, "FluentTTS: Text-dependent fine-grained style control for multi-style tts," *Proc. Interspeech 2022*, pp. 4561–4565, 2022.

[12] Y. Zhang, R. J. Weiss, H. Zen, Y. Wu, Z. Chen, R. Skerry-Ryan, Y. Jia, A. Rosenberg, and B. Ramabhadran, "Learning to speak fluently in a foreign language: Multilingual speech synthesis and cross-language voice cloning," *Interspeech 2019*, 2019.

[13] D. Xin, T. Komatsu, S. Takamichi, and H. Saruwatari, "Disentangled speaker and language representations using mutual information minimization and domain adaptation for cross-lingual tts," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6608–6612.

[14] E. Casanova, J. Weber, C. D. Shulby, A. C. Junior, E. Gölge, and M. A. Ponti, "Yourtts: Towards zero-shot multi-speaker tts and zero-shot voice conversion for everyone," in *International Conference on Machine Learning*. PMLR, 2022, pp. 2709–2720.

[15] H. Cho, W. Jung, J. Lee, and S. H. Woo, "SANE-TTS: Stable And Natural End-to-End Multilingual Text-to-Speech," in *Proc. Interspeech 2022*, 2022, pp. 1–5.

[16] J. Ye, H. Zhou, Z. Su, W. He, K. Ren, L. Li, and H. Lu, "Improving cross-lingual speech synthesis with triplet training scheme," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 6072–6076.

[17] J.-H. Kim, H.-S. Yang, Y.-C. Ju, I.-H. Kim, and B.-Y. Kim, "Crossspeech: Speaker-independent acoustic representation for cross-lingual speech synthesis," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[18] Y. Cao, X. Wu, S. Liu, J. Yu, X. Li, Z. Wu, X. Liu, and H. Meng, "End-to-end code-switched tts with mix of monolingual recordings," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6935–6939.

[19] Y. Cao, S. Liu, X. Wu, S. Kang, P. Liu, Z. Wu, X. Liu, D. Su, D. Yu, and H. Meng, "Code-switched speech synthesis using bilingual phonetic posteriorgram with only monolingual corpora," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7619–7623.

[20] X. Zhou, X. Tian, G. Lee, R. K. Das, and H. Li, "End-to-end code-switching tts with cross-lingual language model," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7614–7618.

[21] S. Manghat, S. Manghat, and T. Schultz, "Normalization of code-switched text for speech synthesis." in *INTERSPEECH*, 2022, pp. 4297–4301.

[22] H.-S. Yang, J.-H. Kim, Y.-C. Ju, I.-H. Kim, B.-Y. Kim, S.-J. Choi, and H.-Y. Kim, "FACTSpeech: Speaking a Foreign Language Pronunciation Using Only Your Native Characters," in *Proc. INTERSPEECH 2023*, 2023, pp. 606–610.

[23] B. J. Choi, M. Jeong, J. Y. Lee, and N. S. Kim, "Snac: Speaker-normalized affine coupling layer in flow-based architecture for zero-shot multi-speaker text-to-speech," *IEEE Signal Processing Letters*, vol. 29, pp. 2502–2506, 2022.

[24] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.

[25] H. Nam and H.-E. Kim, "Batch-instance normalization for adaptively style-invariant neural networks," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

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

[27] AIHub, "Multi-speaker speech synthesis data." [Online]. Available: https://aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&aihubDataSe=data&dataSetSn=542

[28] Y. Koizumi, H. Zen, S. Karita, Y. Ding, K. Yatabe, N. Morioka, M. Bacchiani, Y. Zhang, W. Han, and A. Bapna, "LibriTTS-R: A Restored Multi-Speaker Text-to-Speech Corpus," in *Proc. INTERSPEECH 2023*, 2023, pp. 5496–5500.

[29] 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*. PMLR, 2023, pp. 28 492–28 518.

[30] OpenAI, "Chatgpt: Generative pre-trained transformer for chatbots," *OpenAI Blog*, 2020. [Online]. Available: https://openai.com/blog/chatgpt