

TRIAAN-VC: TRIPLE ADAPTIVE ATTENTION NORMALIZATION FOR ANY-TO-ANY VOICE CONVERSION

Hyun Joon Park*, Seok Woo Yang*, Jin Sob Kim, Wooseok Shin, and Sung Won Han**

School of Industrial and Management Engineering, Korea University, Seoul, Republic of Korea

ABSTRACT

Voice Conversion (VC) must be achieved while maintaining the content of the source speech and representing the characteristics of the target speaker. The existing methods do not simultaneously satisfy the above two aspects of VC, and their conversion outputs suffer from a trade-off problem between maintaining source contents and target characteristics. In this study, we propose Triple Adaptive Attention Normalization VC (TriAAN-VC), comprising an encoder-decoder and an attention-based adaptive normalization block, that can be applied to non-parallel any-to-any VC. The proposed adaptive normalization block extracts target speaker representations and achieves conversion while minimizing the loss of the source content with siamese loss. We evaluated TriAAN-VC on the VCTK dataset in terms of the maintenance of the source content and target speaker similarity. Experimental results for one-shot VC suggest that TriAAN-VC achieves state-of-the-art performance while mitigating the trade-off problem encountered in the existing VC methods.

Index Terms— adaptive attention normalization, any-to-any, siamese loss, voice conversion

1. INTRODUCTION

VC is the task of transforming the voice of the source speaker into that of the target speaker while maintaining the linguistic content of the source speech. Traditional methods require parallel data for training VC models [1, 2] or cannot convert using unseen speakers [3, 4]. For the diverse utilization of VC, researchers have recently studied any-to-any (A2A) and one-shot VC, which can be applied to unseen speakers and require only one utterance of source and target speakers [5–13]. To perform the conversion, they disentangle the utterances into content and speaker representations.

As vector quantization methods, [6, 8] utilized discrete codes as content and the difference between discrete and continuous features as speaker representations. However,

* Equal contribution.

** Corresponding author.

This research was supported by a Korea TechnoComplex Foundation Grant (R2112651, R2112652) and Korea University Grant (K2107521, K2202151). This research was also supported by Brain Korea 21 FOUR.

representing content with discrete codes reduces time relationships, damaging content information. For attention-based conversion methods [10, 12], [10] suggested self-supervised learning features can improve VC performance. Although their results were highly similar to the target speaker characteristics, the conversion method using overly detailed speaker representation biased the results only to speaker similarity. Inspired by image style transfer, [7, 9, 11] adopted Adaptive Instance Normalization (AdaIN) [14] for conversion. [7] used only high-level speaker features for AdaIN, causing results to be biased to speaker similarity. [9] alleviated the problem by exploiting multi-level target speaker features for AdaIN, but AdaIN cannot represent enough speaker characteristics.

Although the previous methods achieved significant improvements in A2A VC, their methods utilized overly detailed or generalized speaker representations; therefore, the conversion results satisfied only one aspect of VC (i.e., maintenance of source content or similarity to the target speaker). This underscores the necessity for a conversion method using core speaker representations to mitigate the trade-off problem.

We propose Triple Adaptive Attention Normalization VC (TriAAN-VC) for non-parallel A2A VC. TriAAN-VC, which is based on an encoder-decoder structure, disentangles content and speaker features. TriAAN block extracts each detailed and global speaker representation from disentangled features and uses adaptive normalization for conversion. As a training approach, siamese loss with time masking is applied to maximize the maintenance of the source content. In A2A one-shot VC, a comparison of results on the VCTK dataset shows that TriAAN-VC achieves state-of-the-art performance in terms of both evaluation metrics, namely, maintenance of source content and similarity to the target speaker.

2. METHOD

2.1. Feature extraction

As [10] suggested Contrastive Predictive Coding (CPC) features [15] contribute to the improvement of VC performance, we adopt CPC features as inputs for the model. We use a pre-trained model [16] to extract CPC features $x \in \mathbb{R}^{H \times T}$ from raw audio $x_{raw} \in \mathbb{R}^t$, where H and T are the hidden size and segment length of x and t is the signal length of x_{raw} .

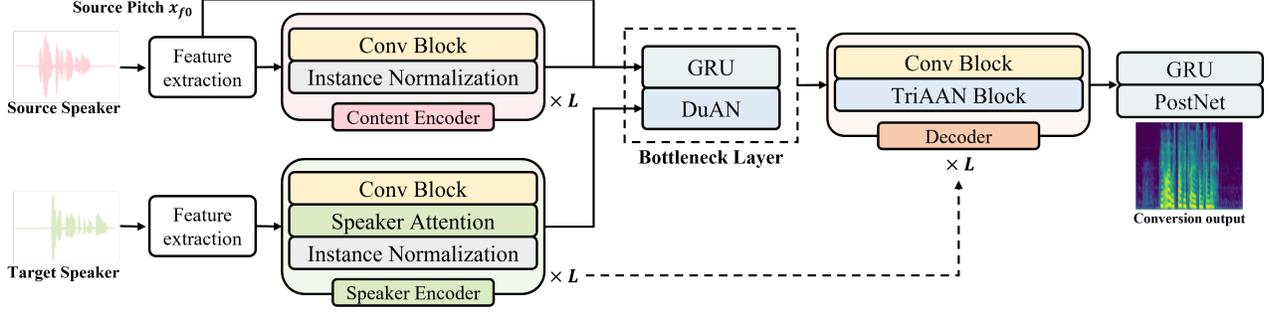


Fig. 1. Overall architecture of TriAAN-VC.

Furthermore, to represent the pitch information of the source speaker, the log fundamental frequency (f0) $x_{f0} \in \mathbb{R}^T$ is extracted by applying the DIO algorithm to x_{raw} , as in [17].

2.2. Encoder and Decoder

As shown in Figure 1, TriAAN-VC comprises two encoders, extracting content and speaker information respectively, and a decoder. The encoders and decoder are connected via a bottleneck layer, and each contains L layers.

Before the encoders, we apply a convolution layer on $x_{c,s} \in \mathbb{R}^{H \times T}$ to expand H to channel size C , where $x_{c,s}$ are the content and speaker inputs after feature extraction.

Encoder. Each encoder layer consists of a convolution block and Instance Normalization (IN). Speaker Attention (SA) is used only in the speaker encoder. The convolution block is designed as a residual block comprising two convolution layers with a kernel size of 3 and a stride of 1.

Bottleneck layer. After the encoder process, f0 of the source speaker $x_{f0} \in \mathbb{R}^T$ is used to represent the pitch. Given $x'_c \in \mathbb{R}^{C \times T}$ is the output of the content encoder, we apply a Gated Recurrent Unit (GRU) layer on the concatenated outputs between x'_c and x_{f0} . Before the decoder, the initial converted representation is generated by applying Dual Adaptive Normalization (DuAN), a conversion method described in Section 2.4, to the content and speaker representations.

Decoder. Each decoder layer contains a convolution block, the same as that of encoders, and TriAAN block. TriAAN block conducts conversion using the content feature from the previous layer and gathered feature maps from the speaker encoder layers. Finally, the outputs of the decoder are refined by GRU layers and PostNet [18] to predict the log mel-spectrogram $\hat{y} \in \mathbb{R}^{M \times T}$, where M is the number of mel bins.

2.3. Speaker Attention

Since AdaIN conversion process utilizes channel-wise statistics of speaker representation, extracting core channel features of speaker is necessary. To achieve it, we modify IN as Time-wise IN (TIN) and design TIN-based Speaker Attention (SA). In contrast to IN, TIN normalizes with the time-wise

mean and standard deviation, preserving channel relations. For SA, we utilize TIN and self-attention [19] as follows:

$$Q = \text{TIN}(x_s)W_q, \quad K = x_sW_k, \quad V = x_sW_v \quad (1)$$

$$\text{Attention}(Q, K, V) = \text{softmax}(QK^T/\sqrt{d})V$$

$x_s \in \mathbb{R}^{T \times C}$ and $W_{q,k,v} \in \mathbb{R}^{C \times C}$ denote speaker features and each weight. Using query information as TIN results, SA emphasizes and preserves the channel relations of speaker features used as speaker information for conversion.

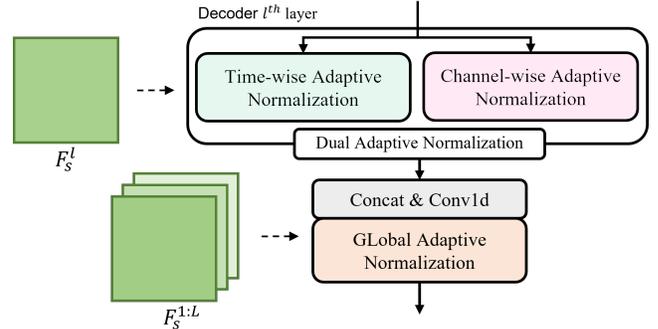


Fig. 2. TriAAN Block

2.4. TriAAN block

As depicted in Figure 2, we design TriAAN block consisting of Dual Adaptive Normalization (DuAN) and GLocal Adaptive Normalization (GLAN) for the conversion process. TriAAN block uses gathered feature maps $F_s^{1:L} \in \mathbb{R}^{T \times C}$ from each l^{th} speaker encoder layer, where $l = 1, 2, \dots, L$. DuAN extracts layer-wise detailed speaker features from F_s^l in dual view (i.e., time and channel) and performs adaptive normalization. By contrast, GLAN uses all feature maps from the speaker encoder to extract global speaker information.

DuAN. Inspired by adaptive attention normalization in image style transfer [20], we design DuAN to extract detailed speaker features and to conduct conversion. DuAN represents attention-based statistics of layer-wise speaker features F_s^l .

Given $x_c \in \mathbb{R}^{T \times C}$ is the content feature from the previous layer and $\mathcal{N}(\cdot)$ is the normalization function, the attention weight $\alpha \in \mathbb{R}^{T \times T}$, attention-weighted mean $M \in \mathbb{R}^{T \times C}$, and variance $Var \in \mathbb{R}^{T \times C}$ are defined as follows:

$$\begin{aligned} Q &= \mathcal{N}(x_c)W_q, \quad K = \mathcal{N}(F_s^l)W_k, \quad V = F_s^lW_v \\ \alpha &= \text{softmax}(QK^\top / \sqrt{d}) \\ M &= \alpha V, \quad Var = \alpha(V \cdot V) - M \cdot M \end{aligned} \quad (2)$$

$W_{q,k,v} \in \mathbb{R}^{C \times C}$ denotes each weight for linear transformation. α , obtained by the normalized feature, represents the similarity between the content and speaker features. Furthermore, Var is calculated using the expectation of variables and the square of the variable expectations. By applying α , the weighted mean and variance contain detailed speaker features, that is per-point statistics. To prevent biased results with excessively detailed speaker features, we take the time-wise average of M and Var , followed by applying a square root on Var to obtain the standard deviation $S \in \mathbb{R}^C$. The converted representation $x'_c \in \mathbb{R}^{T \times C}$, obtained by adaptive normalization and the content feature, is defined as $x'_c = \text{IN}(x_c)S + M$. To perform it in terms of channel and time, we separate the adaptive normalization process depending on $\mathcal{N}(\cdot)$ function (i.e., IN and TIN), making two converted representations.

GLAN. To represent the global speaker information, we utilize all feature maps $F_s^{1:L}$ from the speaker encoder. For the content feature $x_c \in \mathbb{R}^{C \times T}$ in GLAN, we apply a convolution layer to the channel-wise concatenation of two converted representations from DuAN. We obtain layer-wise concatenated means $\mu \in \mathbb{R}^{L \times C}$ and standard deviations $\sigma \in \mathbb{R}^{L \times C}$ from $F_s^{1:L}$, defined as $\mu = [\text{avg}(F_s^1); \dots; \text{avg}(F_s^L)]$ and $\sigma = [\text{std}(F_s^1); \dots; \text{std}(F_s^L)]$. To extract core statistics from global speaker features (i.e., μ and σ) for adaptive normalization, we adopt self-attention pooling [21], which emphasizes important speaker statistics. The attention pooling process used to obtain the weighted mean $\mu' \in \mathbb{R}^C$ and standard deviation $\sigma' \in \mathbb{R}^C$ is as follows:

$$\begin{aligned} \alpha_\mu &= \text{softmax}(\mu W_\mu), \quad \alpha_\sigma = \text{softmax}(\sigma W_\sigma) \\ \mu' &= \text{sum}(\mu \times \alpha_\mu), \quad \sigma' = \text{sum}(\sigma \times \alpha_\sigma) \end{aligned} \quad (3)$$

$\alpha_{\mu,\sigma} \in \mathbb{R}^L_{\mu,\sigma}$ and $W_{\mu,\sigma} \in \mathbb{R}^{C \times C}_{\mu,\sigma}$ denote attention weights and each weight for transformation. Then, adaptive normalization as in DuAN is applied with μ' and σ' for conversion.

2.5. Loss function

To train TriAN-VC, we combine reconstruction loss and siamese loss. Reconstruction loss is $L1$ loss between the ground truth mel-spectrogram $y \in \mathbb{R}^{M \times T}$ and predicted mel-spectrogram $\hat{y} \in \mathbb{R}^{M \times T}$. y is extracted from the raw audio using a mel-spectrogram transformation, and \hat{y} is predicted by the proposed model when input features $x \in \mathbb{R}^{H \times T}$ are CPC features of the raw audio. For siamese loss, $L1$ loss is applied between y and $\hat{y}_{siam} \in \mathbb{R}^{M \times T}$, where \hat{y}_{siam} is predicted by

the model with x augmented by time masking. Given $L1$ loss is $\text{loss}(y, \hat{y}) = \|y - \hat{y}\|_1 / T$, the combined loss is as follows:

$$\mathcal{L} = (\text{loss}(y, \hat{y}) + \text{loss}(y, \hat{y}_{siam})) / 2 + \text{loss}(\hat{y}, \hat{y}_{siam}) \quad (4)$$

By calculating the additional loss with \hat{y}_{siam} , the robustness and consistency of the model can be improved. In particular, since time masking removes content information during training, the loss with the siamese branch makes the model robust for maintaining content information.

3. EXPERIMENTS

3.1. Experimental setup

For comparison, we use the VCTK dataset [22] containing about 400 utterances per 109 speakers. We split the dataset into ratios of 60%, 20%, and 20% for train, validation, and test set, respectively, considering speakers and utterances. For conversion scenarios, we select 20 speakers and generate 600 pairs per Seen-to-Seen (S2S) and Unseen-to-Unseen (U2U) scenarios. After downsampling the audio to 16 kHz, we extract CPC, f0, and log mel-spectrogram features based on a frame size of 25ms, hop size of 10ms, and mel bins of 80.

For training details, we use a batch size of 64, an epoch of 400, and Adam optimizer with a learning rate of 10^{-5} . For model parameters, we take $H = 256$, $C = 512$, and $L = 6$. We adopt a Parallel WaveGAN vocoder [23] pre-trained on the VCTK dataset to convert log mel-spectrograms to waveforms. For comparison, benchmark models are reproduced using their official codes. They are trained with mel-spectrogram features except for S2VC which uses CPC features. The conversion results are available on the demo page.¹

3.2. Evaluation metrics

We adopt objective and subjective measures for evaluation. For objective measures, the models are evaluated in respect of two aspects of VC (i.e, maintenance of source content and similarity to the target speaker). Word Error Rate (WER) and Character Error Rate (CER) are used to evaluate the error rate of scripts between the source and converted utterances. The script of the converted utterances is extracted by a pre-trained Wav2Vec 2.0 [24]. For speaker similarity, we adopt the acceptance rate based on Speaker Verification (SV) model as in [10, 12]. The score is measured by the cosine similarity between the target and converted embedding vectors, extracted by the SV model, and the threshold which is determined based on the equal error rate in the VCTK dataset.

In the subjective evaluation, we conduct Mean Opinion Score (MOS) test for naturalness and similarity. Subjects are asked to assign a score from 1 to 5 after listening to converted utterances or a pair of target and converted utterances for naturalness and similarity evaluation. We perform a test on 15

¹<https://winddori2002.github.io/vc-demo.github.io/>

Table 1. Objective evaluation results on the VCTK dataset for any-to-any one-shot voice conversion which uses one target utterance per sample. SV indicates the acceptance rate over the threshold determined by the equal error rate in the dataset.

Model	Word Error Rate (WER %)			Character Error Rate (CER %)			Speaker Verification (SV %)		
	S2S	U2U	Avg.	S2S	U2U	Avg.	S2S	U2U	Avg.
AUTO-VC [5]	29.96	27.33	28.64	15.98	14.75	15.36	36.00	20.84	28.42
AdaIN-VC [7]	47.33	46.84	47.08	28.16	27.79	27.98	89.34	81.17	85.25
AGAIN-VC [9]	28.89	26.45	27.67	15.65	14.52	15.08	71.33	67.00	69.17
VQVC+ [8]	52.92	53.02	52.97	30.57	31.69	31.13	76.34	55.33	65.83
S2VC [10]	44.99	42.65	43.82	25.88	24.89	25.38	95.34	89.17	92.25
VQMIVC [17]	29.30	28.05	28.67	15.71	15.04	15.37	86.33	35.67	61.00
TriAAN-VC	20.73	22.35	21.54	10.79	11.69	11.24	96.00	89.67	92.83

subjects using randomly selected 20 pairs of utterances for the S2S and U2U scenarios, respectively.

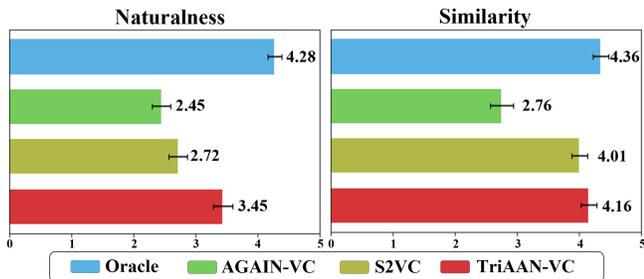


Fig. 3. MOS results with 95% confidence intervals for naturalness and similarity.

3.3. Experimental results

Comparison results. We conducted the experiment for one-shot VC to compare the proposed model with the previous methods using objective and subjective measures. As indicated in Table 1, TriAAN-VC achieved better performance on WER, CER, and SV scores, regardless of conversion scenarios compared to the existing methods which suffered from a trade-off problem of VC. It suggests that the conversion methods using compact speaker features can simultaneously retain both source content and target speaker characteristics.

Figure 3 depicts the average MOS results of the S2S and U2U scenarios, and it includes oracles reconstructed by the vocoder. Similar to the results of objective evaluation, TriAAN-VC demonstrated a slight improvement over S2VC in terms of similarity, which is close to the performance of the oracle. Furthermore, TriAAN-VC outperformed the previous methods in terms of naturalness evaluation, suggesting the proposed model can make relatively unbiased results.

Further experiment. As ablation studies, we analyze the contributions of the proposed components. As listed in rows 1-3 of Table 2, the use of each CPC feature and siamese loss contributed significantly to the performance gain of WER and

SV. In rows 4-6 of Table 2, we excluded one of the components of TriAAN-VC without siamese loss. The results suggested that SA particularly contributed to the improvement of WER, and TriAAN block was the crucial component for the performance gain of SV. Although each component contributed to the performance gain in WER or SV, they also suffered from the trade-off problem, implying all the components are necessary to mitigate the trade-off problem. In addition to one-shot VC, TriAAN-VC was effective in multi-utterance scenarios. Under the multiple utterance setting using more than one target utterance, TriAAN-VC with CPC improved its performance by about 4% and 5% on WER and SV, compared to the one-shot VC results.

Table 2. Results of ablation studies and multi-utterance scenarios. † indicates TriAAN-VC without siamese loss.

Model	WER %		SV %	
	S2S	U2U	S2S	U2U
TriAAN-VC + Mel	27.31	27.07	90.34	88.34
TriAAN-VC + CPC	20.73	22.35	96.00	89.67
TriAAN-VC† + CPC	24.85	26.37	94.67	89.67
- SA	28.16	29.87	93.84	90.17
- GLAN	20.88	21.11	92.50	87.17
- DuAAN	20.58	20.93	90.00	83.00
3-utterance scenario	16.73	18.45	99.00	96.33
5-utterance scenario	16.90	17.82	98.50	98.17

4. CONCLUSION

In this study, we proposed TriAAN-VC for non-parallel A2A VC, which extracts compact speaker features and performs adaptive normalization for conversion. The results of A2A VC on the VCTK dataset indicate that TriAAN-VC achieves outstanding performance, including in multi-utterance scenarios. Unlike previous methods that suffer from a trade-off in VC, TriAAN-VC with siamese loss satisfies two aspects of VC. Finally, ablation studies suggest the necessity of all proposed methods to mitigate the trade-off problem of VC.

5. REFERENCES

- [1] T.Toda, A. W.Black, and K.Tokuda, “Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2222–2235, 2007.
- [2] S.Desai, E. V.Raghavendra, B.Yegnanarayana, A. W.Black, and K.Prahallad, “Voice conversion using artificial neural networks,” in *ICASSP*. IEEE, 2009, pp. 3893–3896.
- [3] H.Kameoka, T.Kaneko, K.Tanaka, and N.Hojo, “Stargan-vc: Non-parallel many-to-many voice conversion using star generative adversarial networks,” in *SLT Workshop*. IEEE, 2018, pp. 266–273.
- [4] T.Kaneko and H.Kameoka, “Cyclegan-vc: Non-parallel voice conversion using cycle-consistent adversarial networks,” in *EUSIPCO*. IEEE, 2018, pp. 2100–2104.
- [5] K.Qian, Y.Zhang, S.Chang, X.Yang, and M.Hasegawa-Johnson, “Autovc: Zero-shot voice style transfer with only autoencoder loss,” in *ICML*. PMLR, 2019, pp. 5210–5219.
- [6] D.-Y.Wu and H.-y.Lee, “One-shot voice conversion by vector quantization,” in *ICASSP*. IEEE, 2020, pp. 7734–7738.
- [7] J.-c.Chou, C.-c.Yeh, and H.-y.Lee, “One-shot voice conversion by separating speaker and content representations with instance normalization,” *arXiv preprint arXiv:1904.05742*, 2019.
- [8] D.-Y.Wu, Y.-H.Chen, and H.-Y.Lee, “Vqvc+: One-shot voice conversion by vector quantization and u-net architecture,” *arXiv preprint arXiv:2006.04154*, 2020.
- [9] Y.-H.Chen, D.-Y.Wu, T.-H.Wu, and H.-y.Lee, “Again-vc: A one-shot voice conversion using activation guidance and adaptive instance normalization,” in *ICASSP*. IEEE, 2021, pp. 5954–5958.
- [10] J.-h.Lin, Y. Y.Lin, C.-M.Chien, and H.-y.Lee, “S2vc: a framework for any-to-any voice conversion with self-supervised pretrained representations,” *arXiv preprint arXiv:2104.02901*, 2021.
- [11] Y.Gu, Z.Zhang, X.Yi, and X.Zhao, “Mediumvc: Any-to-any voice conversion using synthetic specific-speaker speeches as intermedium features,” *arXiv preprint arXiv:2110.02500*, 2021.
- [12] Y. Y.Lin, C.-M.Chien, J.-H.Lin, H.-y.Lee, and L.-s.Lee, “Fragmentvc: Any-to-any voice conversion by end-to-end extracting and fusing fine-grained voice fragments with attention,” in *ICASSP*. IEEE, 2021, pp. 5939–5943.
- [13] Q.Wang, X.Zhang, J.Wang, N.Cheng, and J.Xiao, “Drvc: A framework of any-to-any voice conversion with self-supervised learning,” in *ICASSP*. IEEE, 2022, pp. 3184–3188.
- [14] X.Huang and S.Belongie, “Arbitrary style transfer in real-time with adaptive instance normalization,” in *Proc. ICCV*. IEEE, 2017, pp. 1501–1510.
- [15] A. v. d.Oord, Y.Li, and O.Vinyals, “Representation learning with contrastive predictive coding,” *arXiv preprint arXiv:1807.03748*, 2018.
- [16] M.Riviere, A.Joulin, P.-E.Mazaré, and E.Dupoux, “Unsupervised pretraining transfers well across languages,” in *ICASSP*. IEEE, 2020, pp. 7414–7418.
- [17] D.Wang, L.Deng, Y. T.Yeung, X.Chen, X.Liu, and H.Meng, “Vqmivc: Vector quantization and mutual information-based unsupervised speech representation disentanglement for one-shot voice conversion,” *arXiv preprint arXiv:2106.10132*, 2021.
- [18] Y.Wang, R.Skerry-Ryan, D.Stanton, Y.Wu, R. J.Weiss, N.Jaitly, Z.Yang, Y.Xiao, Z.Chen, S.Bengio, et al., “Tacotron: Towards end-to-end speech synthesis,” *arXiv preprint arXiv:1703.10135*, 2017.
- [19] A.Vaswani, N.Shazeer, N.Parmar, J.Uszkoreit, L.Jones, A. N.Gomez, Ł.Kaiser, and I.Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.
- [20] S.Liu, T.Lin, D.He, F.Li, M.Wang, X.Li, Z.Sun, Q.Li, and E.Ding, “Adaattn: Revisit attention mechanism in arbitrary neural style transfer,” in *Proc. ICCV*. IEEE/CVF, 2021, pp. 6649–6658.
- [21] W.Cai, J.Chen, and M.Li, “Exploring the encoding layer and loss function in end-to-end speaker and language recognition system,” *arXiv preprint arXiv:1804.05160*, 2018.
- [22] C.Veaux, J.Yamagishi, K.MacDonald, et al., “Superseded-cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit,” 2016.
- [23] R.Yamamoto, E.Song, and J.-M.Kim, “Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram,” in *ICASSP*. IEEE, 2020, pp. 6199–6203.
- [24] A.Baevski, Y.Zhou, A.Mohamed, and M.Auli, “wav2vec 2.0: A framework for self-supervised learning of speech representations,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 12449–12460, 2020.