Cycle consistent network for end-to-end style transfer TTS training

Liumeng Xue, Shifeng Pan, Lei He, Lei Xie, Frank K. Soong

Audio, Speech and Language Processing Group (ASLP@NPU), School of Computer Science, Northwestern Polytechnical University, Xi’an, China
Microsoft, China

ARTICLE INFO
Article history:
Received 19 July 2020
Received in revised form 21 February 2021
Accepted 2 March 2021
Available online 16 March 2021

Keywords:
Speech synthesis
End-to-end
Style transfer
Variational autoencoder
Cycle consistent

ABSTRACT
In this paper, we propose a cycle consistent network based end-to-end TTS for speaking style transfer, including intra-speaker, inter-speaker, and unseen speaker style transfer for both parallel and unparallel transfers. The proposed approach is built upon a multi-speaker Variational Autoencoder (VAE) TTS model. The model is usually trained in a paired manner, which means the reference speech is totally paired with the output including speaker identity, text, and style. To achieve a better quality for style transfer, which for most cases is in an unpaired manner, we augment the model with an unpaired path with a separated variational style encoder. The unpaired path takes as input an unpaired reference speech and yields an unpaired output. The unpaired output, which lacks direct ground-truth target, is then successfully constrained by a delicately designed cycle consistent network. Specifically, the unpaired output of the forward transfer is fed into the model again as an unpaired reference input, and after the backward transfer yields an output expected to be the same as the original unpaired reference speech. Ablation study shows the effectiveness of the unpaired path, separated style encoders and cycle consistent network in the proposed model. The final evaluation demonstrates the proposed approach significantly outperforms the Global Style Token (GST) and VAE based systems for all the six style transfer categories, in metrics of naturalness, speech quality, similarity of speaker identity, and similarity of speaking style.

1. Introduction

The goal of speech synthesis is to generate natural speech from text. In recent years, the state of the art text-to-speech (TTS) approaches have been advanced rapidly. The first important one is the advance from hidden Markov models (HMMs) (Tokuda, Yoshimura, Masuko, Kobayashi, & Kitamura, 2000) to neural network (NN) based models, such as fully connected feed-forward network (Ze, Senior, & Schuster, 2013) and long-short term memory (LSTM) (Fan, Qian, Xie, & Soong, 2014). One of the main factors that makes NN-based models outperform HMM-based models is the changing from state- to frame-mapping (Watts, Henter, Merritt, Wu, & King, 2016). To achieve frame mapping, an alignment module is essential to align texts and corresponding acoustic features in the time domain. Besides, a complex text frontend is also necessary to extract various linguistic features. The whole speech synthesis pipeline thus is somewhat heavy. Each module of the pipeline is correlated but trained separately, resulting in errors accumulating along the synthesis pipeline, which hinders TTS performance further improvement. To overcome this barrier, end-to-end models (Li, Liu, Liu, Zhao, & Liu, 2019; Shen et al., 2018; Sotelo et al., 2017; Wang et al., 2017; Wang, Xu, & Xu, 2016), such as the Tacotron family (Shen et al., 2018; Wang et al., 2017), which are based on the sequence-to-sequence mapping paradigm (Sutskever, Vinyals, & Le, 2014) with attention mechanism (Bahdanau, Cho, & Bengio, 2015), are proposed. The end-to-end systems can model alignments between text representations and acoustic features during learning the mapping. Additionally, they use character or grapheme sequence as input, which discards the text analyzer and simplifies the TTS pipelines. With the further help of neural vocoders (Juvela, Bollepalli, Yamagishi, & Alku, 2019; Kumar et al., 2019; Oord et al., 2016, 2018; Ping, Peng, & Chen, 2019; Prenger, Valle, & Catanzo, 2019; Valin & Skoglund, 2019), these models can generate speech with much higher quality waveform than conventional signal-processing based vocoders (Kawahara, Masuda-Katsuse, & De Cheveigne, 1999; Morise, Yokomori, & Ozawa, 2016).

1.1. The style transfer task

With the fast increase of TTS application, there has been a growing need for TTS with more speaking styles, rather than the common neural style, to generate a more human-like or more
delightful speech. To build a TTS voice with multiple speaking styles, one simple way is first collecting recordings of the target speaker in each style and then building a multi-style TTS model. However, it is apparently time-consuming and costly to do so. On the other hand, style transfer provides another way that could be more cost-saving and scalable. It has attracted much interest in recent years (Skerrv-Ryan et al., 2018; Wang et al., 2018; Zhang, Pan, He, & Ling, 2019). Style transfer in TTS aims to transfer the speaking style of a reference speech to the synthesized speech, in a fine-grained or high-level manner. When the text of reference is the same as that to be synthesized, it is possible and usually expected to do a fine-grained transfer, such as transfer the detailed prosodic pattern of the reference to the target (Karlapati et al., 2020; Klimkov, Ronanlki, Rohnke, & Drugman, 2019; Lee & Kim, 2019; Skerry-Ryan et al., 2018). When it is different, the high-level style, such as overall tone, speaking rate, pitch level, and expressivity, can be transferred. Style transfer can be classified into several categories according to whether it is performed across text or speaker, as shown below.

View from text dimension:

- **Parallel transfer**: The text content of the reference audio is the same as that to be synthesized. It is possible to seek a fine-grained transfer.
- **Unparallel transfer**: The text content of the reference audio is different from that to be synthesized. Only high-level transfer can be expected. It could be more frequently used than parallel transfer, without the limitation on text.

View from speaker dimension:

- **Intra-speaker transfer**: The speaker identity of the reference audio is the same as that of the synthesized speech.
- **Inter-speaker transfer**: The speaker identity of the reference audio is different from that of the synthesized speech. It has more potential applications without the limitation for reference speakers. Usually, the inter-speaker transfer is studied only for seen speakers. We would like to call it the unseen speaker transfer which will also be studied in this paper, as defined below.
- **Unseen speaker transfer**: The speaker identity of the reference audio is unseen in training.

When combining the dimension of text and speaker, it will be as many as six categories of style transfer, as listed in 1. In this paper, all six categories will be studied.

1.2. The background

For the research of style transfer, there are two typical models: Global Style Token (GST) based model (Wang et al., 2018) and Variational Autoencoder (VAE) based model (Zhang et al., 2019), where GST or VAE are applied to learn a style embedding which accounts for the speaking style of reference speech and can control the style of synthesized speech. Promising results were reported for intra-speaker style transfer. Recently, there have been a few papers trying to tackle the inter-speaker style transfer (Kulkarni, Colotte, & Jouvet, 2020; Sorin, Shechtman, & Hoory, 2020). In Kulkarni et al. (2020), Kulkarni et al. tried to obtain a better style embedding through VAE followed by an inverse autoregressive flow (IAF). The style embedding is then conditioned to a multi-speaker TTS model to do inter-speaker style transfer. In Sorin et al. (2020), Sorin et al. tried to derive a purified style embedding which is disentangled from text and speaker information for inter-speaker style transfer, through Principal Component Analysis (PCA) on the reference embedding.

So far, all the above models are trained in a paired manner because the reference speech and the target of model output are exactly the same. The paired manner, in the context of style transfer, is embodied by three main dimensions, which are speaker identity, text, and speaking style. When such models are used for style transfer in an unpaired manner, no matter whether for only one or more of the three dimensions, performance degradation is inevitable due to the mismatch between training and inference. To bridge the gap, some works introduce unpaired training into the model (Bian, Chen, Kang, & Pan, 2019; Ma, McDuff, & Song, 2019; Whitehill, Ma, McDuff, & Song, 2020). In Bian et al. (2019), only a paired output is generated as usual. The unpaired output in Ma et al. (2019) and Whitehill et al. (2020), which lacks a ground-truth target to constrain in training, is then constrained by a loss between the style embeddings of the reference speech and the unpaired output speech. This loss, however, is not so solid because it cannot sufficiently guarantee the performance of the detailed output.

1.3. The proposed model

In this work, we focus on style transfer including all six categories listed in Table 1. To make it more general, we assume there is no style label provided. Hence the style is transferred in an unsupervised manner. As aforementioned, the key problem for style transfer, mostly unpaired in at least one dimension between the reference and target output, is the mismatch between the paired training and unpaired inference. In this work, a unified model for style transfer is proposed, including paired path and unpaired path. The model is built upon a multi-speaker VAE based TTS model which only has a paired path. Firstly, it is augmented with an unpaired path with a separated style encoder, along which an unpaired reference is input and an unpaired output is generated, making the unpaired behavior seen during training. The paired and unpaired paths (inputs, outputs) are defined as follows.

- **Paired path**: The reference speech and the output speech share all three dimensions, which are speaker identity, text, and speaking style. Accordingly, the text, reference audio and output are paired text input, reference input and output.
- **Unpaired path**: The reference speech and the output speech differ in at least one of three dimensions, which are speaker identity, text, and speaking style. Accordingly, the text, reference audio and output are unpaired text input, reference input and output.

It is referred to as a base model where the unpaired output has no ground-truth target to constrain. Secondly, a cycle consistent network is designed on top of the above base model, through which the cascaded unpaired output can be constrained by a ground truth target after two times of transfer. To be specific, the unpaired output of the first transfer (forward transfer) is fed into the model again to perform a second transfer (backward transfer). The unpaired output of the backward transfer is expected to reconstruct the original unpaired reference input of the forward transfer, resulting in a cycle consistency. The whole model can therefore be trained effectively.

The proposed model provides a fundamental solution for style transfer, where the unpaired path can be further divided into more branches regarding the degree of mismatching. As we will show later, it is better to use a separate style encoder for each path, to better learn the style representation under different situations. In the work of this paper where multi-speaker rich-style data is used, we start with a focus on the inter-speaker situation for the unpaired path, to get a better transfer performance for inter-speaker style transfer. While it is totally applicable to further expand an unpaired path for intra-speaker unpaired situation (unpaired in text and/or style), which should be more...
capable of intra-speaker unpaired transfer. Objective and subjective evaluation results consistently show that the proposed model is significantly better than GST and VAE based models in intra-speaker, inter-speaker, and unseen speaker style transfer in metrics of naturalness, speech quality, similarity of speaker identity and similarity of speaking style.

The main contributions of this work are summarized as follows:

- We propose a unified framework for style transfer, where different paths are designed to work better for different categories of style transfer, and style label is not needed.
- To the best of our knowledge, for the first time, the cycle consistent network with forward and backward transfer is introduced into style transfer TTS, which makes the unpaired output constrained effectively and the model training algorithmically tractable.
- Comprehensive evaluations are conducted for style transfer, objectively and subjectively with different metrics, which shows the proposed method significantly outperforms two current state-of-the-art approaches, i.e., GST based and VAE based models.

The rest of the paper is organized as follows. Section 2 introduces the end-to-end TTS. Section 3 introduces the related works of style transfer. Section 4 describes the Tacotron2 and two style transfer models: i.e., GST and VAE based models. Section 5 presents the proposed model. Section 6 shows the experimental results. Finally, conclusions are drawn in Section 7.

2. End-to-end TTS

As shown in Fig. 1, the typical parametric TTS pipeline is composed of various complicated modules, including several text analysis modules such as text normalization, lexicon, grapheme to phoneme (G2P) and prosodic analysis. Moreover, separate duration and acoustic models are desired. All modules and models are built separately with different optimization target. The so-called end-to-end TTS significantly simplifies the pipeline by a unified neural network with an attention-based encoder–decoder (AED) structure, where the encoder functions as the text analysis modules and the decoder works like the acoustic model. More importantly, the explicit duration modeling is subtitled by the attention mechanism which learns soft time alignment with the text representation and the acoustic representation. Specifically, in end-to-end TTS with AED structure, the encoder encodes the input text sequence \( x = (x_1, \ldots, x_T) \) into hidden states \( h = (h_1, \ldots, h_T) \):

\[
h_t = \text{encoder}(h_{t-1}, x_t)
\]

Then the decoder generates the current output conditioned on previous prediction at each step:

\[
s_t = \text{decoder}(s_{t-1}, y_{t-1}, c_t)
\]

where \( c_t \) is the context vector calculated by an attention mechanism which allows a decoder to feasibly select encoder hidden states to focus on while generating the output:

\[
c_t = \text{attention}(s_{t-1}, h)
\]

Thus the input text sequence is transformed to output speech sequence \( y = (y_1, \ldots, y_T) \) based on conditional probability \( p(y_1, \ldots, y_T | x_1, \ldots, x_T) \), and the conditional probability can be formulated as:

\[
p(y | x) = \prod_{t=1}^{T} p(y_t | y_1, \ldots, y_{t-1}, x) \tag{4}
\]

\[
p(y_t | y_1, \ldots, y_{t-1}, x) = \mathcal{L}(f(s_t)) \tag{5}
\]

where \( f(.) \) is a fully connected layer. The linear projection is used to predict acoustic features directly based on decoder hidden state \( s \).

In general, works in end-to-end TTS can be categorized into three directions: seeking network structures leading to more natural speech (Li et al., 2019; Shen et al., 2018; Wang et al., 2017) and further simplified pipeline (Donahue, Dieleman, Binkowski, Elsen, & Simonyan, 2020; Weiss, Skerry-Ryan, Battenberg, Marlooyad, & Kingma, 2020), exploring and improving the attention mechanism with more robust speech generation (Chorowski, Bahdanau, Serdyuk, Cho, & Bengio, 2015; He, Deng, & He, 2019) and studying style modeling, control and transfer (Wang et al., 2018; Zhang et al., 2019), expressiveness (Skerry-Ryan et al., 2018) as well as multi-speaker (Gibiansky et al., 2017; Ping et al., 2018).

3. Related works

3.1. Style transfer

The end-to-end TTS model provides a flexible and powerful structure, where many kinds of attributes, such as speaker identity, style, can be conditioned on and influence the corresponding attributes of synthesized speech. For style transfer in TTS, usually a reference speech is needed to provide the target style to transfer. Followed by a reference encoder (Skerry-Ryan et al., 2018), the style embedding can be extracted from the reference speech, which is then used to condition the TTS model.

Skerry-Ryan et al. first extended the Tacotron architecture with a reference encoder so that the model can learn the representation of prosody directly from reference mel spectrogram in an unsupervised fashion (Skerry-Ryan et al., 2018). It was reported that it works well for intra-speaker prosody transfer when reference text and target text are the same or share a very similar structure. Global Style Token (GST) (Wang et al., 2018) further extends the previous work and adds a style token layer to disentangle styles into a set of tokens, in which intra-speaker style transfer with both parallel and unparallel texts are evaluated. To learn a better latent representation of speaker style for style control and transfer, VAE is introduced into Tacotron2 and better performance in intra-speaker parallel and unparallel transfers were reported (Zhang et al., 2019).

Recently, there are a few works with a focus on inter-speaker style transfer. In Kulkarni et al. (2020), Kulkarni et al. followed VAE TTS structure (Zhang et al., 2019) and employed an IAF to get a better style embedding. In Sorin et al. (2020), Sorin et al. followed a similar structure with (Skerry-Ryan et al., 2018) and applied PCA on reference encoder output to get a low dimensional style embedding which is more robust for style control. Different

<table>
<thead>
<tr>
<th>Table 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Six categories of style transfer viewing from the speaker and text dimensions.</td>
</tr>
<tr>
<td><strong>Speaker dimension</strong></td>
</tr>
<tr>
<td><strong>Target text</strong></td>
</tr>
<tr>
<td><strong>Seen speaker</strong></td>
</tr>
<tr>
<td><strong>Non-target speaker</strong></td>
</tr>
<tr>
<td><strong>Unseen speaker</strong></td>
</tr>
<tr>
<td><strong>Non-target speaker</strong></td>
</tr>
</tbody>
</table>

L. Xue, S. Pan, L. He et al. Neural Networks 140 (2021) 223–236
from all the above methods where it is always a totally paired reference speech used, there have been a few papers recently that use multiple reference inputs and/or introduce in an unpaired training (Bian et al., 2019; Ma et al., 2019; Whitehill et al., 2020).

3.2. Multi-reference models

In Bian et al. (2019), multiple references with multiple style encoders are introduced for style control and transfer. Combined with the intercross training strategy, a set of embeddings that independently account for the main attributes of the reference speeches, such as speaker identity and emotion, can be derived. These embeddings are then concatenated together to control the corresponding attributes of the generated speech. However, the style of output speech is always the same as the “style reference” speeches in training, so it is always paired in training.

In Ma et al. (2019), the unpaired path and the unpaired training are introduced to a base model similar to the reference-encoder model (Skerry-Ryan et al., 2018), to address the mismatch between the paired training and unpaired inference. To constrain the unpaired output in training, a style loss represented by the gram matrices differences between the reference and the output, and a latent space loss represented by embeddings difference between the reference and the output, are used.

In Whitehill et al. (2020), an unpaired training strategy is also introduced into a multi-speaker model where two style encodes are incorporated into Tacotron2, for extraction of speaker and style embedding, respectively. The unpaired training is implemented by feeding two different speeches as references, one for speaker embedding and the other for style embedding, and generating an unpaired output. To constrain this unpaired output in training, firstly, the speaker embedding and style embedding of the unpaired output are extracted through the same speaker encoder and style encoder as what is done for the references. Second, the differences of speaker embedding and style embedding between these inputs and outputs are used as the loss.

The proposed method also tries to address the mismatch problem by introducing in an unpaired path. However, it differs from the two most related papers (Ma et al., 2019; Whitehill et al., 2020) in below three folds:

- We propose a unified model for style transfer, which has an explicit paired path and unpaired path with two separate style encoders. As we will show later, separated style encoders are better for the learning of style representations under different situations. However, it is a shared style encoder in Ma et al. (2019).
- The style transfer in the proposed method is in an unsupervised manner where the style label is not needed. While in Whitehill et al. (2020) it is needed.
- In the proposed method, by introducing in a cycle consistent network and performing forward and backward transfer, the cascaded unpaired output can be constrained effectively by a ground truth mel spectrogram target. While it is some kind of style embedding loss between the style reference speech and the unpaired output speech in Ma et al. (2019) and Whitehill et al. (2020), which is very loose.

3.3. Cycle consistency

The idea of cycle consistency is widely used in transfer tasks, such as neural machine translation, image translation, and speech processing. In machine translation, improving translations via “back translation and reconciliation” is a technique used by human translators (Brislin, 1970), as well as by machines (He et al., 2016). In image translation, cycle consistency is widely applied in generative adversarial network (GAN), such as Disco-GAN (Kim, Cha, Kim, Lee, & Kim, 2017), DualGAN (Yi, Zhang, Tan, & Gong, 2017), and CycleGANs (Zhu, Park, Isola, & Efros, 2017). In the speech processing area, CycleGAN is also applied for voice conversion and transfer between normal speech and Lombard speech (Seshadri, Juvela, Alku Räsänen, 2019; Seshadri, Juvela, Yamagishi, Räsänen and Alku, 2019). Moreover, a closed-loop speech chain model is developed (Tjandra, Sakti, & Nakamura, 2017).

For style transfer in TTS, in Whitehill et al. (2020), Whitehill et al. claimed the cycle consistency is used in their work for the unpaired output. However, as stated previously, there is only one transfer in their method. The “cycle consistency” is simply represented by feeding the unpaired output to style encoders again and measuring the difference between style embeddings of reference and the unpaired output. While in the proposed method, to the best of our knowledge, it is the first time that a cycle consistent network with forward and backward transfer is introduced for style transfer TTS, which makes the unpaired output constrained effectively by a ground-truth mel spectrogram target.

4. Tacotron2 and baseline models

In this work, two popular models for style transfer are used as baselines, i.e., GST and VAE based TTS models. The reasons are: (1) they are very representative methods for style transfer; (2) they perform style transfer in an unsupervised manner, while the method in Whitehill et al. (2020) needs style labels; (3) they can be expanded to multi-speaker model easily, in which way inter-speaker style transfer can be realized by feeding a cross-speaker style reference, better than single-speaker model in such task. In the rest of this section, Tacotron2 model will be first introduced, and then two baseline models are described.
4.1. Tacotron2 model

Tacotron2 (Shen et al., 2018) is one of the most popular end-to-end speech synthesis models, which originates from Tacotron and is able to generate natural-sounding speech from text. Both baseline models are built upon Tacotron2. The framework of Tacotron2 is an encoder–decoder model with an attention as shown in Fig. 2. The encoder aims to extract robust hidden representations from character sequences. Firstly, each input character is presented by a 512-dimensional character embedding. Then the embedding sequence is passed through a stack of three convolutional layers that is used to model long-term context. Finally, the output from the last convolutional layer is fed into a bidirectional LSTM layer to produce an encoded hidden representation.

The decoder is an autoregressive recurrent neural network with an attention, generating mel spectrogram from encoder output one frame at a time. The predicted spectrogram of previous frame time is first passed into a prenet including two fully connected layers, which is vital to learn alignments. The attention which consumes the encoder output is used to summarize the full encoded sequence to a fixed-length context vector. Then the context vector is concatenated and passed into a stack of two-layer unidirectional LSTM. The LSTM output and the context vector are concatenated again and fed into two linear projection layers. One is for mel spectrogram prediction, followed by a five-layer convolutional post-net to predict a residual to improve the overall reconstruction. The other is for stop token prediction to determine when to stop the generation process during inference time.

4.2. Global style token based model

Tacotron can only learn an averaged prosody from training data, producing speech with less expressiveness. To generate more expressive speech, Global Style token (GST) based model was proposed (Wang et al., 2018). It is comprised of a reference encoder, style attention, style embedding, and the Tacotron model. The reference encoder, proposed in Skerry-Ryan et al. (2018), extracts prosody representation from a variable-length reference audio clip into a fixed-length reference embedding. Then, the reference embedding is fed into a style token layer, in which it is applied as a query vector to style attention. The attention aims to measure the style similarity between the reference embedding and global style tokens, and produces a set of weights, which stands for the contribution of each token to the reference embedding. The weighted sum of global style tokens is the final style representation of the reference audio, called style embedding. Finally, the style embedding is added with encoder output to be used as a style condition on the Tacotron to generate speech with a specific style. In our work, we replace the Tacotron with Tacotron2 in the GST model and augment a speaker look-up table to transfer style across speakers, using as one baseline.

4.3. Variational autoencoder based model

Variational Autoencoder (VAE) is a powerful structure to explicitly model latent variable that accounts for the generation of data samples. The latent state of a speaker, when speaking in a certain style, acts in a similar role as the latent variable does in VAE. More specifically, latent representation of speaking style can be inferred from a reference audio by variational inference and then used as style condition of synthesized speech. Following this idea, VAE is introduced into Tacotron2 to learn the latent style representation for style control and transfer (Zhang et al., 2019). It has been shown that the introduction of VAE to the TTS model can learn the latent representation of speaking style in a continuous space in an unsupervised manner. The VAE based TTS model consists of two components: a recognition network and a Tacotron2 TTS model. The recognition network is used to encode the reference audio into a fixed-length vector of latent representation (or latent variables $z$, which stands for style representation). Tacotron2 converts the combination of encoder state and latent representation to synthetic speech with the speaking style from the reference audio. In this paper, we refer to the recognition network as a variational style encoder and describe its structure in Section 5.3. The original VAE based TTS model is a single speaker model. It is augmented with a speaker look-up table to be a multi-speaker model in this work and used as another baseline.

5. The proposed model

5.1. Overall structure

The overall structure of the proposed cycle consistent network is illustrated in Fig. 3. It has two stages: stage 1 forward transfer and stage 2 backward transfer. Each stage has two style reference inputs, and the multi-speaker VAE based TTS model learns to generate two speeches with the same text content and speaker identity, but with two different styles from two references. At stage 1, $X$ is the paired output that is the reconstructed result of the paired reference $X$, and $Z$ is the unpaired output that transfers style from the unpaired reference $Y$. At stage 2, $Y$ is used as the paired reference and $Y'$ is the reconstructed paired output. $Z$ from stage 1 is used as the unpaired reference and $Y'$ is the unpaired output, which transfer the style of $Z$ back to $Y$. Through the forward and the backward transfer, cycle consistency is formed on the unpaired path ($Y-Z-Y'$). Therefore, the whole unpaired
path, with the unpaired output Z included, can be optimized using ground-truth target via back propagation. To independently learn style representations from paired and unpaired references, considering the intrinsic difference of the two paths, two separated style encoders are adopted. The detailed model architecture is illustrated in Fig. 4. And the model hyperparameters are listed in Appendix Table A.8. In the rest of this section, we will describe the details of the proposed model, including the unpaired path, separated style encoders, and cycle consistent network.

5.2. Unpaired path

Generally, for the TTS model with reference encoder, the style reference and the output are always paired in training, such as GST and VAE TTS models. However, it has a mismatch when doing unpaired transfer in inference as mentioned previously, especially for inter-speaker style transfer. To address this issue, we augment an unpaired path on top of the multi-speaker VAE based Tacotron2 model. As shown in the left part of stage 1 in Fig. 4, the paired reference (X) and the paired output (X’) have the same target text, target speaker identity, and style, forming a paired path and working just like the traditional style transfer model. Additionally, the unpaired path, along which the unpaired reference input (Y) (which, e.g., has different speaker identity, text content, and style from the paired reference) is fed into the model, and an unpaired output (Z) that has the target text, target speaker identity, and the style of unpaired reference is generated. In this way, both paired and unpaired transfer behaviors are exposed in training. Specifically, the paired path is designed for paired transfer, and the unpaired path is for unpaired transfer.

The whole process of style transfer TTS for any of the two paths is described as follows. Firstly, speaker embedding is produced via a speaker look-up table (LUT) with a target speaker’s ID as input, and style embedding is derived from the variational style encoder with a style reference as input. Secondly, the text encoder output is added with the style embedding and concatenated with the speaker embedding. Finally, the combined result is passed into attention and decoder to generate speech output. Since the two paths share the same attention and decoder, and to speed up the training, we merge the two combined results (each for one path) into one batch before passing them into the decoder, and then split the batch into two outputs for paired output and unpaired output after decoding.

5.3. Separated style encoders

We adopt the variational style encoder to obtain the style representation from a style reference speech as described in Section 4.3. From the view of autoencoder, i.e., from reference speech to reconstructed speech conditioned on text, it is the recognition model of VAE. From the view of text-to-speech, it is a variational encoder, which learns the style embedding from reference speech and conditions the text-to-speech process. Following the variational principle (Kingma & Welling, 2014), we have:

\[ \log p_\theta(o) = KL[q_\phi(z|o) \parallel p_\theta(z|o)] + \ell(\theta, \phi; o) \]

(6)

where \( o \) is the observed dataset, \( z \) is the unobserved continuous random latent variables, and \( \ell(\theta, \phi; o) \) is the evidence lower bound (ELBO). Considering the additional condition of text \( x \) here, the ELBO can be reformulated as below:

\[ ELBO = E_{z \sim q_\phi(z|x)}[\log p_\theta(z|x)] - KL(q_\phi(z|x) \parallel p_\theta(z)] \]

(7)

The expectation term plays the role of a decoder which reconstructs \( o \) via decoding the latent variables \( z \) and the input text \( x \). In this work, \(-E_{z \sim q_\phi(z|x)}[\log p_\theta(z|x)]\) is referred to as reconstruction loss, and \( KL(q_\phi(z|x) \parallel p_\theta(z)]\) is referred to as KL loss.

As illustrated in Fig. 5, the style encoder consists of a reference encoder and followed by two fully connected (FC) layers with linear activation. First, the reference encoder encodes the style information from a variable-length reference speech to a fixed-dimension reference embedding. The architecture and hyperparameters of reference encoder are the same as Skerry-Ryan et al. (2018), including six 2-D convolutional layers with batch normalization followed by a 128-unit Gate Recurrent Unit (GRU) layer and tanh activation. Then two FC layers produce the mean and standard deviation of the latent style variable \( z \).
The reparameterization trick is used here to make the gradient backpropagation easy in training. Once the latent variable $z$ is obtained, it is passed through a FC layer to produce style embedding, making the dimension of style embedding equal to the text encoder output (character embedding). The style embedding is then added with text encoder output. The result is further concatenated with speaker embedding and consumed by a location-sensitive attention to converts encoded sequence to a fixed-length context vector for each decoder step.

In our work, two separate style encoders are adopted for paired and unpaired references, instead of a shared one as in Ma et al. (2019). We argue that with two separate style encoders used, each of them can work better with different preferences than a shared and averaged one. More specifically, the style encoder for the paired style reference is encouraged to learn speaker- and content-dependent style embedding, while the style encoder for the unpaired style reference is encouraged to learn speaker- and content-independent style embedding. As shown later in the experiment part, the performance by using two style encoders is better than that using a shared one for style transfer.

5.4. Cycle consistent network

At stage 1, through the paired path and unpaired path with two variational style encoders, paired output and unpaired output are generated. We refer to the multi-speaker VAE based model with paired and unpaired paths as the base model. The paired output ($X'$) that is totally the same as the paired reference ($X$) in text, speaker identity, and style, can be optimized directly by the target (paired reference). However, the unpaired output ($Z$) that is expected to be the same as the unpaired reference ($Y$) in style only, cannot be directly optimized because no ground-truth target is available. To solve this problem, a cycle consistent network is delicately designed. As shown in the right part of stage 2 in Fig. 4, we feed the unpaired output ($Z$) of stage 1 back to the base model as a new unpaired reference and feed the original unpaired reference ($Y$) as a new paired reference. Similar to stage 1, two outputs are generated. One is the paired output ($Y''$) corresponding to the paired reference, the other is the unpaired output ($Y'$). After this back transfer, the unpaired output ($Y''$) is expected to be totally the same as the original unpaired reference ($Y$) of stage 1 in text, speaker identity, and style, and can be optimized effectively by target (the unpaired reference of stage 1). Hence, a cycle consistency is formed on the unpaired path ($Y' - Z' - Y''$).

For the paired output at each stage, a reconstruction loss can be used directly. For the unpaired output of stage 2, a cycle consistency loss can be used, which will further constrain the unpaired output of stage 1 by gradient back-propagation along the unpaired path. The total loss $L$ of the proposed model is comprised of stage 1 loss $L_{stage1}$ and stage 2 loss $L_{stage2}$:

$$L = L_{stage1} + L_{stage2}$$

\[L_{stage1} = w_1 \cdot L_{rec} + L_{stop} + w_2 \cdot L_{kl} \]

\[L_{stage2} = w_1 \cdot L_{rec} + L_{stop} + w_3 \cdot L_{kl} \]

where $L_{rec}$ and $L_{stop}$ are reconstruction loss and stop token loss for the paired path. $L_{kl}$ are KL losses for two variational style encoders in paired and unpaired paths. $L_{ccl}$ and $L_{stop}$ are cycle consistency loss and stop token loss for the unpaired path. $w_1$, $w_2$ and $w_3$ are loss weights for the reconstruction loss, KL loss and cycle consistency loss, respectively.

It is worth mentioning that the cycle consistent network is only used in the training phase. In the inference phase, only the base model is used to generate style transfer speech.

6. Experiments

6.1. Experimental setup

Corpus We intend to use an open-source multi-speaker rich-style data for experiments, to facilitate the reproduction. The most appropriate open-source data we could find for this work is LibriTTS (Zen et al., 2019), a multi-speaker audio-book data. This data contains various animated or emotive speaking styles, read by more than 2000 speakers, which is ideal for a solid evaluation including intra-speaker, inter-speaker, and unseen speaker style transfer over a big number of target speakers.

The train-clean-100 and train-clean-360 subsets of the LibriTTS is used in this work. 45 speakers are selected from the two subsets, where each one has more than 300 utterances to ensure the characteristic of each speaker can be well learned. The total number of the selected corpus is 15,468 utterances. The data is further split into two parts: 15,208 utterances for the training set, in a total of 17.76 h, and 270 utterances for the test set including all 45 speakers (6 utterances per speaker). All audio clips are down sampled to 16 kHz and 80-dimensional mel spectrograms are extracted with frame shift 12.5 ms and frame length 50 ms.

Training stage Character sequence is used as text input and 80-dimensional mel spectrogram is predicted as target. 128-dimensional speaker embedding obtained from a speaker look-up table (LUT) is used as the speaker identity condition. Two mel spectrograms, which are randomly selected with different speaker identity, are used as the paired and unpaired references.

As mentioned previously, we intend to firstly focus on the interspeaker situation for the unpaired path in this work. Regarding the property of training data, the two references differ not only in speaker identity, but also in text and style. In this way, the paired and unpaired paths are actually induced to perform intra-speaker parallel transfer and inter-speaker unparallel transfer in training, respectively.

The decoding of the paired path is always in teacher forcing manner in training, which is the same for the unpaired path of stage 2. As to the decoding of the unpaired output of stage 1, first, full inference is tried. However, the training fails to converge. Other strategies are then tried including scheduled sampling, semi sampling, and teacher forcing, where the ground truth information of the corresponding paired path in stage 1 is all involved. It is found that using teacher forcing can generate the most stable result and semi sampling is close to teacher forcing when the weight of ground truth information is high. Finally, teacher forcing is used, which, according to the evaluation results, still shows obvious quality improvement objectively and subjectively. But we believe there is room here for further improvement. Since the two outputs share the same attention and decoder and to speed up the training, the combined encoder outputs of paired and unpaired paths are merged into one combined batch and fed to the attention. After decoding, it is split apart into corresponding paired and unpaired outputs.

The training of the proposed model starts from a pre-trained multi-speaker VAE based model, to ease the convergence. The KL annealing strategy used in Zhang et al. (2019) is also applied in our task to solve the collapse problem of KL loss. Due to

\[\text{The dataset is available at http://www.openslr.org/60/} \]
the limitation of computation resources and for fair comparison, all different models are trained on 4 GPUs with batch size 16, and each model is trained with about 200k steps using Adam optimizer (Kingma & Ba, 2015).

**Inference stage** At inference time, the base model of stage 1 is used to generate speech. Two 80-dimensional mel spectrograms are fed into the model as references. To match the situation in training, the paired reference that has the same speaker identity is used as the target speaker to perform intra-speaker style transfer, while the unpaired reference that has a different speaker identity is used to perform inter-speaker style transfer. In terms of synthesized text, it can be the same as the text transcription of reference for parallel transfer, or be different for unparallel transfer. Under the condition of the same speaker embedding and two different style embeddings, two mel spectrograms are generated. To reconstruct waveform from mel spectrogram, we use the WaveNet (Oord et al., 2016) as a neural vocoder. The WaveNet used in our work is firstly trained with Microsoft TTS data, which consists of 149 speakers and 500 utterances for each speaker, in a total of 80 hours, and then refined using the 45-speaker data.

### 6.2. Evaluation metrics

Two objective metrics and three subjective metrics are used in the evaluations.

**Objective metrics** For objective evaluation, to measure style similarity between the synthesized speech and reference speech. Pearson correlation coefficient (Benesty, Chen, Huang, & Cohen, 2009) is calculated on three phone-level statistics, including long fundamental frequency (LF0), energy, and duration, which are three main prosody features related to style (Chiang, Hung, Yeh, Liao, & Pan, 2019). The higher correlation means the higher style similarity. To get phone-level statistics, forced alignment for generated speech and reference speech are conducted to get phone boundaries. Then phone-level mean values of the prosody features are computed from frame-level prosody features according to phone boundaries. Finally, the correlations between the two phone-level statistics sequences of the synthesized speech and reference speech are measured. This metric is only measured for parallel transfer because there is no target reference for unparallel transfer.

The other objective metric is Fréchet Audio Distance (FAD) (Kilgour, Zuluaga, Roblek, & Sharifi, 2019), which is proposed to measure the quality of music enhancement. It has a high correlation with human perception and is first used to evaluate the quality of synthesized emotional speech in Jia et al. (2019). We use it to measure the speech quality for both parallel and unparallel style transfers, since FAD is a reference-free metric. The lower FAD score means the higher speech quality.

**Subjective metrics** For subjective evaluation, three types of tests are conducted, including comparison mean opinion score (CMOS), ABX preference test in style similarity and speaker similarity, and mean score opinion (MOS). CMOS is used to make a comparison in naturalness between two voices, ranging from −3 to 3. Generally, a positive score means the new voice is better than the baseline voice, while a negative score means it is worse. ABX preference test is used to measure which voice is perceived more similar to a reference, in speaking style or speaker identity. Higher preference means more similarity is perceived. 5-scale MOS is used to measure voice naturalness in a non-comparison manner. In CMOS and ABX preference tests, each case is judged by 9 English native listeners. In the MOS test, each case is rated by 15 English native listeners. In all tests, 50 utterances across 16 speakers are randomly selected for evaluations.

### 6.3. Ablation tests

In this section, we evaluate the effectiveness of three main components of the proposed model, including unpaired path, cycle consistent network, and separated style encoders. The models studied here are designed with a gradual augmentation of the three components. First, starting from a multi-speaker VAE based model, an unpaired path is augmented, forming a model with two paths. The style encoder is shared so far. Then, a cycle consistent network is further augmented, forming a model with a shared style encoder and a cycle consistent network. Finally, the shared style encoder is further separated, forming a model with separated style encoders and a cycle consistent network, i.e., the proposed model. The four models are referred to as VAE, OneStyEncNoCycle, OneStyEncCycle, and TwoStyEncCycle hereafter for simplicity. In training, the loss function of TwoStyEncCycle is shown in Eqs. (8)–(10). While for OneStyEncCycle, the only difference is there is only one KL loss term in each of the two transfer stages. While for model OneStyEncNoCycle, only the paired losses of stage 1 (including reconstruction loss, stop token loss and KL loss) are used.

The objective metrics of the four models are measured first. Then, subjective evaluations are conducted between every two neighboring models along the augmentation line. In ablation tests, only inter-speaker style transfer from unpaired output is considered which is more challenging. We believe a similar result can be found for intra-speaker style transfer as well.

#### 6.3.1. Objective evaluations

For objective evaluations, correlation and FAD are measured, as shown in the inter-speaker columns of Tables 2 and 3. In correlation test, with the augmenting of the components, the LF0 correlation of resulting model keeps increasing, with 0.388 for VAE, 0.415 for OneStyEncNoCycle (unpaired), 0.446 for OneStyEncCycle (unpaired), and 0.528 for TwoStyEncCycle (unpaired). While for correlations of energy and duration, the changes are small. For FAD test, it keeps decreasing for both parallel and unparallel transfers, with the augmenting of the components. The above results objectively indicate the effectiveness of the three components, in terms of better pitch similarity and speech quality.

#### 6.3.2. Subjective evaluations

In subjective evaluations, three groups of comparison experiments are conducted, each of which involves two models — with or without one component. Specifically, they are: (1) **VAE vs OneStyEncNoCycle**, (2) **OneStyEncNoCycle vs OneStyEncCycle**, (3) **OneStyEncCycle vs TwoStyEncCycle**. Preference test and CMOS test are used in these evaluations.

**VAE vs OneStyEncNoCycle** This comparison is to verify the effectiveness of the unpaired path. Preference results are shown in Fig. 6. As we can see, by adding the unpaired path, the speaker similarity gets slightly improvement in parallel transfer and slightly degradation in unparallel transfer. For style similarity, it obtains more preferences in both parallel and unparallel transfers. In CMOS test, OneStyEncNoCycle (unpaired) is significantly better than VAE in unparallel transfer with 0.322, and on par in parallel transfer, as shown in Table 4.

By adding the unpaired path, the unpaired transfer behavior is introduced into training, which should bring benefit for inter-speaker style transfer. This is proved by the above improvements in speaker similarity, style similarity and naturalness. It should be noted, without an effective constraint on the unpaired output, the original difference in training loss between the two models, is just the KL loss term for the unpaired style latent variable in model OneStyEncNoCycle. However, a simple KL regularization for the unpaired path does be helpful. Actually, in mode VAE, the style
encoder is only trained by paired data, unwanted information leakage is inevitable, such as speaker identity and content information. When feeding in an inter-speaker reference in synthesis, such leaked information will degrade the quality. While in model **OneStyEncNoCycle**, the additional KL regularization from the unpaired path can prevent the shared style encoder to be fully biased by the paired path and hence reduce the leakage. The effect is verified by the above test results. Meanwhile, the improvement is also limited, with the only introduction of unpaired path. **OneStyEncNoCycle** vs **OneStyEncCycle** This comparison is to verify the effectiveness of the cycle consistent network. For preference, as shown in Fig. 7, the two models have comparable preferences of style similarity in unparallel transfer and **OneStyEncNoCycle** (unpaired) is more preferred in parallel transfer. For speaker similarity, **OneStyEncCycle** (unpaired) obtains more preference no matter in parallel or unparallel transfer. CMOS results in Table 4 show **OneStyEncCycle** (unpaired) is more natural in both transfers.

In training of model **OneStyEncNoCycle**, the unpaired path only contributes to the style encoder with a corresponding KL loss, which is much weaker than the reconstruction loss from the paired path. The shared style encoder is therefore still biased to the paired path somehow, with a tendency of content-dependent style information learning. As a result, **OneStyEncNoCycle** performs better in parallel transfer. In contrast, the shared style encoder is trained by both paths in **OneStyEncCycle**, which leads to an averaged style information learning and therefore no special strength for both parallel and unparallel transfers. However, one benefit of the cycle network here is its capability in keeping better quality (including naturalness and speaker identity) in inter-speaker style transfer. As a matter of fact, when there is no constraint on the unpaired output during training as in **OneStyEncNoCycle**, either the style embedding contains certain speaker identity information or the combination of speaker embedding and the style embedding from another speaker’s speech is very unseen, both of which tend to degrade the quality of the unpaired output in inter-speaker style transfer. On the contrary, with the help of a cycle network, the unpaired path can be trained effectively with a ground-truth target, yielding better quality in both naturalness and speaker identity.

**OneStyEncCycle** vs **TwoStyEncCycle** This comparison is to verify the effectiveness of the separated style encoders. Preference test is shown in Fig. 8. As we can see, **TwoStyEncCycle** (unpaired) obtains much more preferences of style similarity for both parallel and unparallel transfers. It also has more preferences in speaker similarity, especially in parallel transfer. In CMOS test, as shown in Table 4, it shows **TwoStyEncCycle** (unpaired) is more natural in both transfers, especially in the parallel one with CMOS 0.407.

The above tests consistently show the model with separated style encoders performs better than a shared one. Actually, it

### Table 2

Correlation results for parallel transfer.

<table>
<thead>
<tr>
<th>Models</th>
<th>Intra-speaker</th>
<th>Inter-speaker</th>
<th>Unseen-speaker</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LF0</td>
<td>Energy</td>
<td>Duration</td>
</tr>
<tr>
<td>GST</td>
<td>0.501</td>
<td>0.831</td>
<td>0.746</td>
</tr>
<tr>
<td>VAE</td>
<td>0.561</td>
<td>0.845</td>
<td>0.765</td>
</tr>
<tr>
<td>OneStyEncNoCycle unpaired</td>
<td>/</td>
<td>/</td>
<td>/</td>
</tr>
<tr>
<td>OneStyEncCycle unpaired</td>
<td>/</td>
<td>/</td>
<td>/</td>
</tr>
<tr>
<td>TwoStyEncCycle paired</td>
<td>0.569</td>
<td>0.839</td>
<td>0.786</td>
</tr>
<tr>
<td>TwoStyEncCycle unpaired</td>
<td>0.545</td>
<td>0.839</td>
<td>0.771</td>
</tr>
</tbody>
</table>

For intra-speaker style transfer, p-value is calculated between the paired output of the proposed model (TwoStyEncCycle paired) and each other models (including TwoStyEncCycle unpaired).

For inter-speaker style transfer, p-value is calculated between the unpaired output of the proposed model (TwoStyEncCycle unpaired) and each other models (including TwoStyEncCycle paired).

* p-value < 0.05.

** p-value < 0.01.

### Table 3

FAD results for parallel and unparallel transfers.

<table>
<thead>
<tr>
<th>Models</th>
<th>Intra-speaker</th>
<th>Inter-speaker</th>
<th>Unseen-speaker</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LF0</td>
<td>Energy</td>
<td>Duration</td>
</tr>
<tr>
<td>GST</td>
<td>1.558</td>
<td>1.598</td>
<td>1.734</td>
</tr>
<tr>
<td>VAE</td>
<td>1.462</td>
<td>1.581</td>
<td>1.547</td>
</tr>
<tr>
<td>OneStyEncNoCycle unpaired</td>
<td>/</td>
<td>/</td>
<td>/</td>
</tr>
<tr>
<td>OneStyEncCycle unpaired</td>
<td>/</td>
<td>/</td>
<td>/</td>
</tr>
<tr>
<td>TwoStyEncCycle paired</td>
<td>1.269</td>
<td>1.324</td>
<td>1.273</td>
</tr>
<tr>
<td>TwoStyEncCycle unpaired</td>
<td>1.150</td>
<td>1.268</td>
<td>1.245</td>
</tr>
</tbody>
</table>

Table 4

CMOS results for different models compared with base models.

<table>
<thead>
<tr>
<th>Baseline voice</th>
<th>New voice</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parallel</td>
<td>Unparallel</td>
</tr>
<tr>
<td>----------------</td>
<td>-----------</td>
</tr>
<tr>
<td>VAE</td>
<td>OneStyEncNoCycle unpaired</td>
</tr>
<tr>
<td>0.027</td>
<td>0.322</td>
</tr>
<tr>
<td>OneStyEncNoCycle unpaired</td>
<td>OneStyEncCycle unpaired</td>
</tr>
<tr>
<td>0.096</td>
<td>0.142</td>
</tr>
<tr>
<td>OneStyEncCycle unpaired</td>
<td>TwoStyEncCycle unpaired</td>
</tr>
<tr>
<td>0.407</td>
<td>0.156</td>
</tr>
</tbody>
</table>

Fig. 6. Preference results of VAE and the model using a shared style encoder without a cycle network (OneStyEncNoCycle unpaired) on inter-speaker style transfer.

Fig. 8. Preference results of VAE and the model using a separated style encoder (TwoStyEncCycle unpaired) on inter-speaker style transfer.
is natural that separated style encoders can learn style information from paired and unpaired references independently and with different focus. Specifically, the style encoder of unpaired path is encouraged to learn speaker- and content-independent style information, given the training strategy described in Section 6.1. Apparently, it is able to perform better than a shared style encoder, which mixes up the two paths in learning of style representation, in inter-speaker style transfer.

To summarize, the above objective and subjective evaluations prove the effectiveness of the three main components of the proposed model \((\text{TwoStyEncCycle})\), where the unpaired path with a separated style encoder can learn style information that is independent of the speaker identity and text content, while the cycle network guarantees the success of better naturalness and higher speech quality in inter-speaker style transfer.

### 6.4. Final evaluations

In this section, we first examine the strength of each path of the proposed model in style transfer. Then, the comparisons between the proposed model and two baselines, i.e., GST and VAE, are conducted, for intra-speaker, inter-speaker and unseen speaker style transfer.

#### 6.4.1. Paired path vs unpaired path

As described in the training stage of Section 6.1, given the training strategy, the paired and unpaired paths are actually induced to perform intra-speaker parallel transfer and inter-speaker unparallel transfer in training, respectively. It is interesting to study how they perform for other types of style transfer. To do this experiment, the same style references are fed into the style encoders. More specifically, the same paired references that contain target speaker identity are fed as paired and unpaired references to perform intra-speaker style transfer, and the same unpaired references are fed to perform inter-speaker style transfer. For parallel transfer, the text to synthesis is the same as the reference, while it is different for unparallel transfer. By feeding the same references, the two paths work in the same way but with different style embeddings extracted from two separated style encoders.

**Intra-speaker style transfer** For intra-speaker style transfer, the paired path \((\text{TwoStyEncCycle paired})\) gets slightly higher correlation in LF0 and duration, and the same in energy, as shown in Table 2. Preference results in Fig. 9 show the two paths have comparable preference of speaker similarity, but the paired path obtains more style similarity preferences. From the FAD results and MOS results presented in Tables 3 and 5, we can see though the unpaired path gets slightly higher speech quality, the paired path still gets better naturalness.

It is reasonable that the paired path works better for intra-speaker parallel transfer, which exactly matches its training. While for intra-speaker unparallel transfer, it is different from the training situation for both paired and unpaired paths. It looks like the style embedding from a speaker- and content-dependent style encoder (the paired path) is more effective than that from a speaker- and content-independent style encoder (the unpaired path), for intra-speaker unparallel transfer.

**Inter-speaker style transfer** The correlation results in Table 2 show the unpaired path \((\text{TwoStyEncCycle unpaired})\) achieves lower duration correlation with statistical significance. This can be attributed to the using of paired target for teacher forcing training on unpaired path at stage 1. However, the unpaired path has a much higher LF0 correlation and comparable energy correlation, and the difference of LF0 correlation is statistically significant.

In preference results as presented in Fig. 10, the unpaired path has slightly more preferences in parallel transfer. For unparallel transfer, we expected more preferences could be obtained, while actually only 1.33% and 1.45% more preferences are obtained for...
Table 5
MOS results.

<table>
<thead>
<tr>
<th>Models</th>
<th>Intra-speaker</th>
<th>Inter-speaker</th>
<th>Unseen-speaker</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Parallel</td>
<td>Unparallel</td>
<td>Parallel</td>
</tr>
<tr>
<td>Recording</td>
<td>4.34 ± 0.08</td>
<td>4.35 ± 0.08</td>
<td>4.38 ± 0.08</td>
</tr>
<tr>
<td>GST</td>
<td>3.20 ± 0.09</td>
<td>3.10 ± 0.08</td>
<td>2.95 ± 0.08</td>
</tr>
<tr>
<td>VAE</td>
<td>3.24 ± 0.09</td>
<td>3.24 ± 0.08</td>
<td>3.28 ± 0.08</td>
</tr>
<tr>
<td>TwoStyEncCycle</td>
<td>3.54 ± 0.08</td>
<td>3.59 ± 0.08</td>
<td>3.48 ± 0.07</td>
</tr>
<tr>
<td>paired</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TwoStyEncCycle</td>
<td>3.48 ± 0.08</td>
<td>3.52 ± 0.08</td>
<td>3.47 ± 0.07</td>
</tr>
<tr>
<td>unpaired</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6
MOS gains of the proposed model compared with two baselines.

<table>
<thead>
<tr>
<th>Baseline</th>
<th>Intra-speaker</th>
<th>Inter-speaker</th>
<th>Unseen-speaker</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Parallel</td>
<td>Unparallel</td>
<td>Parallel</td>
<td>Unparallel</td>
</tr>
<tr>
<td>GST</td>
<td>0.34</td>
<td>0.49</td>
<td>0.62</td>
<td>0.62</td>
</tr>
<tr>
<td>VAE</td>
<td>0.30</td>
<td>0.35</td>
<td>0.29</td>
<td>0.37</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.45</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.56</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.52</td>
</tr>
<tr>
<td>GST</td>
<td></td>
<td></td>
<td></td>
<td>0.56</td>
</tr>
<tr>
<td>VAE</td>
<td></td>
<td></td>
<td></td>
<td>0.39</td>
</tr>
</tbody>
</table>

Table 7
Preference gains of the proposed model compared with two baselines.

<table>
<thead>
<tr>
<th>Preference type</th>
<th>Baseline</th>
<th>Intra-speaker</th>
<th>Inter-speaker</th>
<th>Unseen-speaker</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Parallel</td>
<td>Unparallel</td>
<td>Parallel</td>
<td>Unparallel</td>
</tr>
<tr>
<td>Style similarity</td>
<td>GST</td>
<td>25.54%</td>
<td>19.63%</td>
<td>29.67%</td>
<td>16.78%</td>
</tr>
<tr>
<td></td>
<td>VAE</td>
<td>16.08%</td>
<td>18.91%</td>
<td>18.55%</td>
<td>16.78%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>33.34%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>14.26%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>23.20</td>
</tr>
<tr>
<td>Speaker similarity</td>
<td>GST</td>
<td>19.74%</td>
<td>21.98%</td>
<td>41.45%</td>
<td>32.59%</td>
</tr>
<tr>
<td></td>
<td>VAE</td>
<td>15.36%</td>
<td>19.26%</td>
<td>32.11%</td>
<td>31.97%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>30.36</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>26.17</td>
</tr>
</tbody>
</table>

6.4.2. Intra-speaker style transfer

The paired path of the proposed model is used for intra-speaker style transfer evaluation against baselines, because this path is trained with paired speaker identity and proved to perform better in intra-speaker transfer previously. Specifically, in inference, the intra-speaker style transfer is performed by feeding a reference to the paired path which has the same speaker identity as the target speaker. For intra-speaker parallel transfer, the text to synthesis is the same as the reference of the paired path, while it is different for intra-speaker unparallel transfer.

The correlation results in Table 2 show that, compared with GST, the proposed model TwoStyEncCycle (paired output) has much higher LF0 and duration correlation and slightly higher energy correlation. Compared with VAE, it has slightly lower LF0 and energy correlation but much higher duration correlation. The differences of duration correlation between the proposed model and two baselines are statistically significant. The preference test results are illustrated in Figs. 11 and 12. It indicates TwoStyEncCycle (paired output) outperforms baseline models in terms of speaker and style similarity for both parallel and unparallel transfers. In addition, much lower FAD scores and much higher MOS scores (0.3+) are obtained by TwoStyEncCycle (paired output), as shown in Tables 3 and 5.

Objective and subjective test results show the paired output of the proposed model performs significantly better than baselines. 

4 samples can be found at [https://lmxue.github.io/style-transfer-journal/index.html](https://lmxue.github.io/style-transfer-journal/index.html).
in intra-speaker style transfer, in terms that it generates speech with higher quality and better naturalness while maintaining more similar speaker identity and speaking style.

6.4.3. Inter-speaker style transfer

The unpaired output of the proposed model is used for the evaluations of inter-speaker style transfer, because the unpaired path is right designed for that and proved to perform better in inter-speaker transfer previously. More specifically, in inference, the inter-speaker style transfer is performed by feeding a reference to the unpaired path which has a different speaker identity from the target speaker. For inter-speaker parallel transfer, the text to synthesis is the same as the reference of the unpaired path, while it is different for inter-speaker unparallel transfer.

The correlation results in Table 2 show that, compared with GST, the proposed model TwoStyEncCycle (unpaired output) gains much higher correlation in LFO and duration with statistical significance and slight higher correlation in energy. Compared with VAE, it also gains much higher LFO correlation with statistical significance and slightly higher correlation in energy and duration. Preference test results are presented in Figs. 13 and 14. It shows the proposed model is always more preferred in speaker similarity and style similarity for both parallel and unparallel transfers. Meanwhile, the lowest FAD scores are obtained by the proposed model, as shown in Table 3. Furthermore, the MOS results in Table 5 show it is significantly more natural, with about 0.6+ over GST and about 0.3 over VAE.

Testing results objectively and subjectively demonstrate the proposed model is effective in transferring style across speakers, in terms of better capturing style from inter-speaker reference, better retaining speaker identity in transferred speech, higher speech quality, and better naturalness.

6.4.4. Unseen speaker style transfer

We further study the performance for unseen speaker (the speaker of reference is unseen in training) style transfer. The unpaired path of the proposed model is used to do unseen speaker style transfer, because the unseen speaker is also a kind of unpaired situation.

As shown in Table 2, from seen inter-speaker to unseen speaker, the correlation result gets obvious regression for all three models. However, there are still obvious gains of the proposed model TwoStyEncCycle over the baselines, with statistical significance for almost all prosody dimensions but energy against the VAE baseline. A similar trend can be found in the FAD result, as shown in Table 3. In preference tests shown in Figs. 15 and 16, the proposed model has more than half preferences in parallel test and almost half preferences in unparallel test, which is much better than both baselines. In addition, it also obtains the highest MOS scores with about 0.5 gain over the baselines, as shown in Table 5. The above results show that even when the style reference is from an unseen speaker, the proposed model still keeps an obvious advantage over the baselines, indicating its good generalization capability.

MOS and preference gains obtained by the proposed model over the baselines are further summarized in Tables 6 and 7. It is easy to find the highest MOS gain over GST baseline occurs in inter-speaker style transfer (0.62 by average), and its unseen-speaker style transfer against VAE baseline (0.5 by average). For preference, the highest gain of style transfer over two baselines is on unseen speaker, and the highest gain of speaker similarity is on inter-speaker style transfer. Overall, for all six types of style transfer, the proposed model gets 0.52 and 0.39 MOS gains, about 23% and 17% style similarity, and about 30% and 17% speaker similarity preference gains over GST and VAE, respectively.

7. Conclusion

We propose a cycle consistent network based end-to-end TTS for speaking style transfer, including intra-speaker, inter-speaker, and unseen speaker style transfer for both parallel and unparallel transfers. The proposed model is built upon an end-to-end, multi-speaker VAE TTS model. In addition to the paired path, along which a paired style reference is input and a paired output is
generated, we augment an unpaired path with a separated style encoder, along which an unpaired reference is input and an unpaired output is generated. In this work, we feed the unpaired speaker identity reference as an unpaired reference to make the behavior of inter-speaker style transfer seen during training. Moreover, a cycle consistent network is designed to make the unpaired output, i.e., inter-speaker style transfer output, constrained effectively. Ablation test shows the effectiveness of three main components of the proposed model, including the unpaired path, separated style encoders and cycle consistent network. Final evaluation demonstrates the proposed model performs significantly better than two current state-of-the-art approaches, i.e., GST and VAE based models for all the six style transfer categories, in naturalness, speech quality, and similarity in speaking style and speaker identity.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Key R&D Program of China under Grant No. 2020AAA0108600.

Appendix. Model hyperparameters

<table>
<thead>
<tr>
<th>Table A.8</th>
<th>Hyperparameters of the proposed model. “conv1d-k-c-BN-ReLU” denotes 1-D convolution with width k, output channels c, and with batch normalization and ReLU activation. “conv2d-(k,k)-(s,s)-c-BN-ReLU” denotes 2-D convolution with kernel size (k,k), stride (s,s), output channels c and with batch normalization and ReLU activation. FC-u-sigmoid stands for fully-connected layer with units u and sigmoid activation.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spectral analysis</td>
<td>pre-emphasis: 0.97; frame length: 50 ms; frame shift: 12.5 ms</td>
</tr>
<tr>
<td>Character embedding</td>
<td>512-D</td>
</tr>
<tr>
<td>Speaker embedding</td>
<td>128-D</td>
</tr>
<tr>
<td>Style embedding</td>
<td>512-D</td>
</tr>
<tr>
<td>Mel spectrogram</td>
<td>80-D</td>
</tr>
<tr>
<td>Reduction factor (r)</td>
<td>1</td>
</tr>
<tr>
<td>Text encoder</td>
<td>3 convolution layers: <code>conv1d-5-512-BN-ReLU</code> → <code>conv1d-5-512-BN-ReLU</code> → <code>conv1d-5-512-BN-ReLU</code> Bidirectional LSTM: 256 cells</td>
</tr>
<tr>
<td>Decoder</td>
<td>2-layer pre-Net: <code>FC-256-ReLU</code> → Dropout(0.5) → <code>FC-256-ReLU</code> → Dropout(0.5) 2 LSTM layers: 1024 cells 5 conv layer post-net: 4-layer <code>conv1d-5-512-BN-ReLU</code> → Dropout(0.5) → <code>conv1d-5-512-BN</code> → Dropout(0.5) Linear projection: <code>FC-80</code> Stop token projection: <code>FC-1-sigmoid</code></td>
</tr>
<tr>
<td>Style encoder</td>
<td>6-layer <code>conv2d</code>: <code>conv2d-(3,3)-(2,2)-32-BN-ReLU</code> → <code>conv2d-(3,3)-(2,2)-32-BN-ReLU</code> → <code>conv2d-(3,3)-(2,2)-64-BN-ReLU</code> → <code>conv2d-(3,3)-(2,2)-64-BN-ReLU</code> → <code>conv2d-(3,3)-(2,2)-128-BN-ReLU</code> → <code>conv2d-(3,3)-(2,2)-128-BN-ReLU</code> GRU: 128 cells tanh activation: FC-128-tanh</td>
</tr>
</tbody>
</table>

References

He, M., Deng, Y., & He, L. (2019). Robust sequence-to-sequence acoustic modeling with stepwise monotonic attention for neural TTS. In INTERSPEECH.


