Interspeech 2024
1-5 September 2024, Kos, Greece

Pre-training Neural Transducer-based Streaming Voice Conversion for Faster Convergence and Alignment-free Training

Hiroki Kanagawa, Takafumi Moriya, Yusuke Ijima

NTT Corporation, Japan
hiroki.kanagawa@ntt.com

Abstract

Seq2seq-based voice conversion (seq2seq VC) can model without pre-aligning speech lengths, but non-monotonic attention matrices may cause unnatural VC. The VC-T, a streaming VC based on neural transducer, ensures monotonic alignments, outperforming seq2seq VC. However, VC-T demands guiding alignments and time-consuming training due to tensor computations. The guiding alignments are generated from manually annotated phoneme labels with labor-intensive efforts, and they are essential for eliminating improbable paths, thus stabilizing VC-T training. This work proposes a two-stage VC-T training pipeline for fast convergence: 1) VC-T pre-training to learn probable paths, which form a matrix, optimized by L1 loss, 2) Fine-tuning refines the pre-trained VC-T, outputting a probable tensor from the start. This enables VC-T training without the guiding alignment. Experiments show our pipeline achieves superior streaming VC while significantly reducing training time compared to conventional VC-T.

Index Terms: streaming voice conversion, neural transducer, pre-training

1. Introduction

Voice conversion (VC) is a technique that modifies the speaker characteristics from a source speaker to a target speaker while preserving the linguistic content. Recent neural network based approaches can be categorized into non-parallel and parallel VCs. Non-parallel VC methods enable learning without parallel data between the source and target speakers. Most of these approaches separate the input speech into content and speaker information, training to reconstruct the input speech from these separated components [1–9]. However, since the reconstruction loss is calculated from the input and predicted speech of the same length, this makes VC challenging for speaking rate conversion, which is an important factor of speaker characteristics. An approach combining automatic speech recognition (ASR) and text-to-speech (TTS) enables speaking rate conversion but requires audio/text pairs for training [10]. Therefore, our focus is on parallel VC, which can easily convert speaking rates and can also be learned using only audio data.

Parallel VC provides more stable training with less data compared to non-parallel VC because it utilizes the same speech content for training between the source and target speakers. Early approaches using dynamic time warping (DTW) for frame alignment removed duration information, making speaking rate conversion impossible, similar to many non-parallel VC methods [11–14]. In contrast, modern sequence-to-sequence VC models that employ an attention-based encoder-decoder architecture account for differences in both duration and spectral characteristics [15–17]. Since these approaches are independent of DTW, they enable voice conversion with modified speaking rates. However, if monotonic attention is not achieved, the resulting VC may contain speech drops and repetitions, leading to unnatural conversion. This issue is especially acute for streaming operations, since future contexts are not available.

To overcome this drawback of seq2seq VC, VC-T [18] introduced a neural transducer (RNNT) [19] into VC framework. RNNT is ideal for streaming ASR [20] and TTS [21, 22], as it guarantees the monotonic alignment generation. The VC-T offers enhanced naturalness compared to conventional seq2seq VC while preserving linguistic content, thanks to the robust alignment prediction of RNNT. However, its training demands a substantial amount of memory because it involves a three-dimensional tensor, whereas seq2seq VC training uses a two-dimensional matrix. This requires time-consuming operations during training and makes VC-T difficult to handle. Moreover, stable VC-T training heavily depends on guiding alignments, which are generated from manually annotated phoneme labels with laborious efforts, to avoid improbable paths.

To address these issues, we propose the pre-training approach for VC-T. This approach 1) ensures faster convergence, and 2) enables alignment-free training. This idea stems from SimpleFlat [23], which pre-trains all model parameters using frame-level cross-entropy loss. Our proposed method first aligns the source and target speech sequences to have the same length using DTW. Since the input and output lengths match, the VC-T outputs during pre-training form a 2D matrix. This allows pre-training all VC-T parameters faster using frame-level L1 loss, thanks to reduced memory demands. Then, we finetune the pre-trained VC-T model with the modified VC-T loss, using only unaligned speech spectra. At the start of fine-tuning, the pre-trained model can transduce the alignments between speakers. Thus, this step does not require labor-intensive alignment data. Moreover, our pre-trained VC-T converges much faster than training a VC-T from scratch. On one-to-one VC tasks, the most basic setup of parallel VC, our fine-tuned streaming VC-T initialized by the proposed pre-training, outperforms conventional seq2seq VC and the original VC-T by both objective and subjective measures. Surprisingly, this improvement required only one-tenth of the original training steps[1].

2. VC-T: Voice conversion based on RNNT

Figure 1 is a schematic diagram of VC-T [18] training. The VC-T model is composed of the source and target speaker's speech encoders, and the joiner. The source and target speaker's speech encoders convert the source and target speaker's speech into their respective hidden representations. These are then fed

[1] Sample audios are available here:
https://hkanagawa.github.io/interspeech2024vct-pt/

10.21437/Interspeech.2024-771

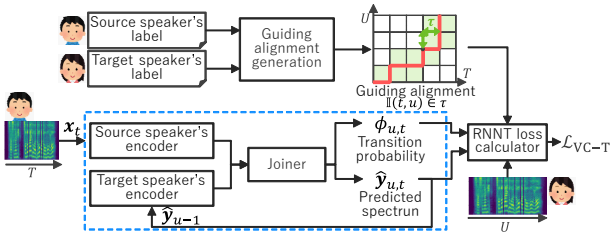


Figure 1: *Overview of VC-T.*

to the joiner, which outputs $[\hat{\boldsymbol{y}}_{t,u}, \phi_{t,u}]$: the predictions of the spectrum, $\hat{\boldsymbol{y}}_{t,u}$, and its transition probability, $\phi_{t,u}$. Note that $\phi_{t,u}$ is pre-activated by a sigmoid function. The forward variable $\alpha_{t,u}$ for VC-T loss is calculated as follows;

$$\alpha_{t,u} = \alpha_{t-1,u}\phi_{t-1,u} + \alpha_{t,u-1}\left(1 - \phi_{t,u-1}\right), \tag{1}$$

where T and U are the frame lengths of the source spectra $\boldsymbol{X} \in \mathbb{R}^{T\times D}$ and target spectra $\boldsymbol{Y} \in \mathbb{R}^{U\times D}$, respectively. t and u indicate the respective indices. Thus, the VC-T loss is given by:

$$\mathcal{L}_{\text{VC-T}} = \sum_{t=1}^{T}\sum_{u=1}^{U} \mathbb{I}\left\{(t,u)\in\tau\right\}\alpha_{t,u}\left(1-\phi_{t,u}\right)\left|\boldsymbol{y}_{u+1} - \hat{\boldsymbol{y}}_{t,u}\right|, \tag{2}$$

where $\boldsymbol{y}_{u+1} \in \mathbb{R}^D$ is the $u+1$-th frame of target speaker's spectrum. $\hat{\boldsymbol{y}}_{t,u} \in \mathbb{R}^D$ is the t-th and u-th predicted spectrum. To stabilize training, we omit improbable paths using an indicator function, $\mathbb{I}\left\{(t,u)\in\tau\right\}$, to identify whether the index (t,u) is within the guiding alignment with bandwidth τ. The actual guiding alignment ($\tau = 0$) is depicted in Fig. 3 (a).

The guiding alignment helps with VC-T training, and VC-T consistently generates monotonic alignment, resulting in natural VC [18]. However, the joiner's output $\hat{\boldsymbol{\mathcal{Y}}} \in \mathbb{R}^{T\times U\times(D+1)}$ forms a three-dimensional tensor, resulting in severe memory consumption during training. This is due to the different lengths of the source and target audio. Moreover, while $\mathbb{I}\left\{(t,u)\in\tau\right\}$ is generated from the guiding alignment, its generation requires laboriously annotated phoneme labels.

3. Proposed VC-T training pipeline

In this section, we first explain our pre-training proposal for faster convergence, followed by the proposed guiding alignment-free fine-tuning. Then, we also introduce a multi-frame prediction scheme for faster training and accurate prediction.

3.1. Pre-training step for VC-T

VC-T requires simultaneous learning of spectra and alignments, which increases the training difficulty resulting in more training steps being needed steps for the convergence. To address these concerns, we propose VC-T pre-training, inspired by Simpleflat [23], for the ASR task. Simpleflat performs entire RNNT network pre-training, and its parameters are optimized with cross-entropy loss using the RNNT's predictions and coarse frame-level alignments, which represent the pre-aligned sequence lengths of input and output. This yields better initialization than a random approach and results in improved ASR performance. In this work, we modified Simpleflat for VC-T.

Figure 2 overviews our pre-training proposal for VC-T. We align the input and output spectra using DTW based on their spectral similarity to ensure they have the same length, resulting in $\boldsymbol{X}^{\text{align}} \in \mathbb{R}^{U'\times D}$ and $\boldsymbol{Y}^{\text{align}} \in \mathbb{R}^{U'\times D}$, where U' is the aligned spectrum frame length. Thus, the VC-T joiner output

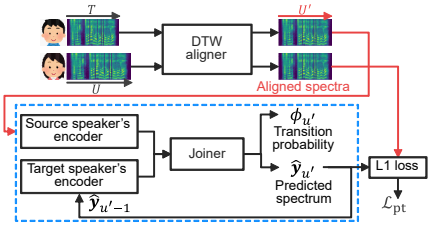


Figure 2: *Overview of the proposed pre-training phase for VC-T.*

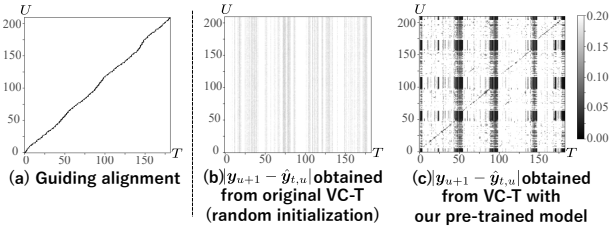


Figure 3: *The actual alignments of (a) ground-truth guiding alignment, (b), and (c) obtained from VC-T with random initialization or pre-trained parameters, respectively. Note that (b) and (c) illustrate the magnitudes of $|\boldsymbol{y}_{u+1} - \hat{\boldsymbol{y}}_{t,u}|$ before training with each VC-T loss (i.e., the 0-th epoch).*

forms a two-dimensional matrix, $\hat{\boldsymbol{Y}}^{\text{align}} \in \mathbb{R}^{U'\times D}$, during pre-training, which allows us to optimize it using a simple L1 loss as follows:

$$\mathcal{L}_{\text{pt}} = \sum_{u'=1}^{U'} \left|\boldsymbol{y}_{u'}^{\text{align}} - \hat{\boldsymbol{y}}_{u'}^{\text{align}}\right|, \tag{3}$$

where $\boldsymbol{y}_{u'}^{\text{align}}$ and $\hat{\boldsymbol{y}}_{u'}^{\text{align}}$ are u'-th spectra of $\boldsymbol{Y}^{\text{align}}$ and $\hat{\boldsymbol{Y}}^{\text{align}}$, respectively. This pre-training ensures that VC-T can output a better spectral prediction at the start of fine-tuning phase, leading to faster convergence and easing training complexity.

In Figs. 3 (b) and (c), we show the magnitudes of $|\boldsymbol{y}_{u+1} - \hat{\boldsymbol{y}}_{t,u}|$ obtained from VC-T's output with either random initialization or pre-trained parameters, respectively. Figure 3 (b) reveals an alignment resulting for randomly initialized parameters, indicating the absence of any meaningful path of at this stage. Consequently, original VC-T training requires additional assistance, i.e., the guiding alignment shown in Fig. 3 (a), to establish a probable path. In contrast, when employing pre-trained VC-T models, the resulting alignment can effectively capture the probable path, appearing as a diagonal line in Fig. 3 (c). Each dense square on the grid in Fig. 3 (c) represents the pause interval. As we can see, the diagonal line closely follows the guiding alignment in Fig. 3 (a), highlighting the effectiveness of our proposed pre-training. Importantly, our pre-training method does not rely on explicit guidance, such as manually annotated phoneme labels.

3.2. Fine-tuning phase for pre-trained VC-T

We also introduce an enhanced VC-T loss that does not require the guiding alignment. The fine-tuning phase adopts the pre-trained VC-T (in §3.1) as the initialized parameters and optimizes it with a modified VC-T loss, similar to Eq. (2) using the unaligned spectra, i.e., $\boldsymbol{X}$ and $\boldsymbol{Y}$. The modified VC-T loss is defined as follows;

$$\mathcal{L}'_{\text{VC-T}} = \sum_{t=1}^{T}\sum_{u=1}^{U} \alpha_{t,u}\left(1-\phi_{t,u}\right)\left|\boldsymbol{y}_{u+1} - \hat{\boldsymbol{y}}_{t,u}\right|. \tag{4}$$

Equation (2) needs $\mathbb{I}\left\{(t,u)\in\tau\right\}$ which makes VC-T training more complicated due to the use of manually annotated

phoneme labels. On the other hand, as we can see, Eq. (4) eliminates the term $\mathbb{I}\{(t,u) \in \tau\}$, and thus it is obviously much simpler than original VC-T training.

3.3. Multi-frame prediction

Beside pre-training, we further reduce computational complexity by introducing multiple frame prediction into VC-T, which has traditionally involved in conventional acoustic modeling [15, 24, 25]. The reason is that shorter series make the mapping problem easier.

Multi-frame prediction concatenates the source and target speakers' spectra every N frames, yielding $\boldsymbol{X}' \in \mathbb{R}^{\frac{T}{N} \times ND}$ for the source and $\boldsymbol{Y}' \in \mathbb{R}^{\frac{U}{N} \times ND}$ for the target speaker. Thus, the VC-T output forms $\hat{\boldsymbol{\mathcal{Y}}}' \in \mathbb{R}^{\frac{T}{N} \times \frac{U}{N} \times ND}$. This results in a significantly reduced trellis size, which is approximately $\frac{1}{N^2}$ times smaller. Shorter series do not always make the mapping problem easier, and thus the tuning N involves a trade-off. In this paper, we investigate the effectiveness of multi-frame prediction for VC-T.

4. Experiments

4.1. Setup

For training VC models, we used multi-speaker speech data from three professional Japanese narrators, one male and two females. The sampling rate was 22.05 kHz. Each speaker had 1000 parallel utterances, forty were used as the evaluation set (about 3.3 minutes) and the rest as the training data (about 2.9 hours). We used a 80-dimensional log mel-spectrograms as the acoustic features. The analysis frame shift was 12.5 ms. Note that these are exactly same setups as used in [18].

All evaluations were conducted only with streaming VC, since [18] mentioned that streaming VC-T performs comparable to its offline one. Three types of one-to-one VC models were trained for each method: homo-gender conversion (F2→F1), hetero-gender ones (M→F1, and F1→M). As the conventional seq2seq VC, we used ConvS2S-VC [16] with convolutional gated linear units with residual connections (ConvGLU) [26]. To run this in streaming mode, the source and target speaker's speech encoders and decoders were built using six causal ConvGLU blocks with 256 channels. We configured VC-T, the source and target speakers' encoders, with the same number of ConvGLUs channels. Its joiner has three blocks consisting of a linear layer and layer normalization with 256 hidden units and ReLU. These VC models have approximately the same number of model parameters 12.4 and 10.7 million, respectively. For the indicator design of Eq. (2) described in § 2, the guiding alignment derived from the manually annotated phoneme labels was used and τ was set to 1. All of these VC models were optimized in 1000k steps with a batch size of 32 by Adam [27] with a fixed learning rate of 1×10^{-4} using an Intel Xeon Gold 6338 CPU 2.0GHz, and NVIDIA Tesla A100 GPU 80GB. For waveform generation, the speaker-independent HiFi-GAN (v1) [28] was used. We trained it in 1500k steps with the same optimizer setup as VC models, using data from our 1,050 in-house Japanese speakers (about 312.9 hours), including VC training data.

For objective evaluations, we used word error rate (WER), mel-cepstrum distortion (MCD), and F0 root mean square error (RMSE) to cover the three aspects of reproduction, linguistic content, spectral, and prosodic, respectively. WER was calculated for the recognition hypothesises obtained from Whisper (large) [29]. MCD was obtained by converting the mel-spectrogram from VC into a 40-dimensional mel-cepstrum and then aligning the sequence length with that of the target speaker natural speech by DTW. For the F0 RMSE calculation, we first extracted the F0 from the converted speech using the Swipe algorithm [30]. Then, after aligning it with the target speaker's extracted F0 using the DTW paths from the MCD calculation, the RMSE across both voiced segments was calculated.

Table 1: *Elapsed time to train Original VC-T for 500k steps and WERs of each VC output obtained from all evaluation data. N denotes the number of multi-frame predictions. The numbers in parentheses represent the processing time per 1k steps.*

No. of multi-frame N	Elappsed time for 500k steps [day]	WER [%]
1	99.1 (287.8 min. / k steps)	21.4
2	44.1 (126.9 min. / k steps)	18.6
4	24.9 (71.6 min. / k steps)	17.8
6	17.9 (51.7 min. / k steps)	30.7

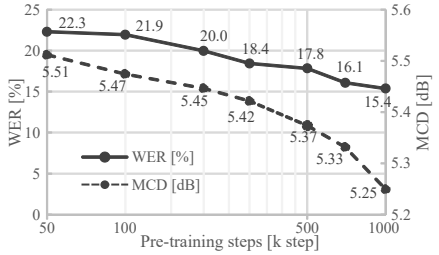


Figure 4: *WERs and MCDs of our pre-trained VC-T for each pre-training step.*

4.2. Parametric study of multi-frame predictions

First, we investigate the best hyperparameter, N, for multi-frame prediction. Table 1 shows WER variation of each VC-T model with $N = 1, 2, 4, 6$ and its elapsed time. As we can see, the training time for 500k steps becomes shorter as N is increased. However, WERs improved up to $N = 4$ but worsened markedly at $N = 6$. Thus, it is evident that simply increasing N is not an appropriate approach. In this work, four-frame concatenation, denoted as VC-T with $N = 4$, achieved the best WER, indicating that this configuration preserves linguistic information most effectively. Actually, by setting $N = 4$ to reduce memory usage, we could update the model every four accumulations, calculating gradients with eight utterances in each batch of 32. Then, the training time for 500k steps was reduced from the original 24.9 days to just 6.2 days. Hereafter, we adopted $N = 4$ for all VC-T models.

4.3. Effectiveness of our proposed VC-T pre-training

We evaluated the performance of the pre-trained VC-T without fine-tuning. The transition probability within VC-T output was not trained during pre-training, but it is assumed that the source and target speakers have the same speaking rate. Therefore, for inference we estimated the target speaker's spectrum from the VC with the source spectrum input that was forced to transition by sequentially incrementing both frame indices, t and u. Figure 4 depicts the WERs and MCDs of the pre-trained VC-T models at each pre-training step. We can see that both WERs and MCDs improved as the pre-training progressed. In particular, the performance of the pre-trained VC-T, even with only 1.7 days of pre-training for the 500k steps, achieved competitive results comparable to the VC-T model trained from scratch using the original VC-T loss equation (Eq. (2)). This impressive result was achieved while significantly reducing the training time from the 6.2 days discussed in §4.2 to only 1.7 days (3.6x faster). We emphasize that our pre-trained VC-T was not fine-tuned and did not use the manually annotated phoneme labels,

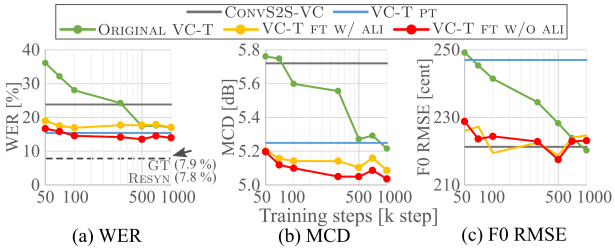


Figure 5: *WER, and MCD, and F0 RMSE comparison with that of* ConvS2S-VC *for each VC-T training step. The WERs are calculated not only for the VC methods, but also for the original speech (*GT*) and the analytic re-synthesized one by HiFi-GAN (*Resyn*).*

which are needed for guiding alignment creation. For the subsequent experiments, we used the 1000k-step pre-trained VC-T, which took 3.4 days to train.

4.4. Overall objective evaluations

We evaluated each method via multiple objective metrics. All compared methods were streaming VCs: ConvS2S-VC, the original VC-T (Original VC-T), pre-trained VC-T described in § 4.3 (VC-T pt), VC-T using VC-T pt as initial model (VC-T ft w/ ali), and also its variant without guiding alignment (VC-T ft w/o ali)[2]. Only the results for 1000k steps are shown for ConvS2S-VC and VC-T pt . The other VC-Ts provided results for each training step to confirm the effectiveness of our proposed pre-training. Note that WERs are calculated not only for the speech converted by VC, but also for the original speech (GT) and the analytic re-synthesized one by HiFi-GAN (Resyn) so as to understand the upper limit performance of the ASR model used.

Figure 5 displays objective evaluation metrics for VC-T at various steps. Original VC-T improved in all metrics as steps increase but took 500k steps to surpass ConvS2S-VC's WER. VC-T pt and 50k steps of VC-T ft w/ ali and VC-T ft w/o ali surpass both in WER and MCD. In terms of these metrics, VC-T ft w/ ali and VC-T ft w/o ali stabilized around 100k and 500k steps, respectively, indicating effective initial values from our pre-training (VC-T pt). While VC-T ft w/ ali converged to Original VC-T's WER, VC-T ft w/o ali outperformed all VCs in terms of WER. This indicates that our pre-training helps train VC-T even without guiding alignment. While guiding alignment enhanced training stability, it assumed a diagonal mapping between source and target speaker phonemes, acting as a constraint on finding the best paths. Thus, VC-T ft w/ ali doesn't guarantee optimal alignment. The F0 RMSE of VC-T pt is less favorable, likely due to non-trainable alignments via external DTWs, negatively impacting F0 prediction. However, alignment-trainable VCs (all except VC-T pt) performed well.

4.5. Subjective evaluations

We subjectively evaluated the naturalness of the converted speech, including Resyn, using a mean opinion score (MOS) on a five-point scale from 5: very natural to 1: very unnatural. The speaker similarity of converted speech to that of the target speaker was also compared to that of ground truth using degradation MOS (DMOS) on a five-point scale from 5: very similar to 1: very dissimilar. Fifteen listeners participated in these tests,

[2]Training original VC-T without guiding alignment resulted in gradient vanishing and learning failure.

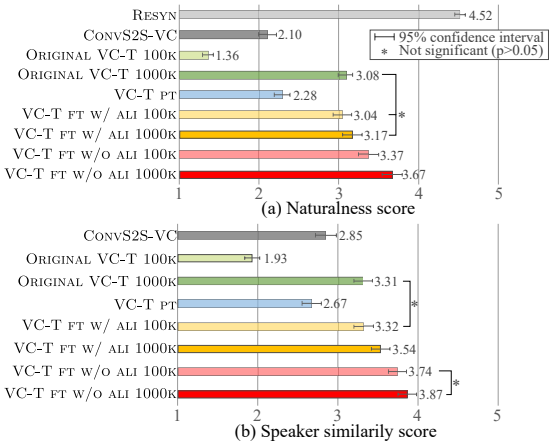


Figure 6: *Subjective evaluation results.*

with six randomly chosen utterances for each VC's gender setting, resulting in 162 utterances for naturalness evaluation and 144 for speaker similarity one across all methods. The comparison methods are the same as for § 4.4. With the exception of VC-T pt, we prepared two variations of VC-T with 100k- and 1000k-steps to demonstrate the convergence speed of our proposed method.

Figure 6 (a) presents the naturalness scores, with the average scores shown due to space constraints. All of VCs scored lower in naturalness compared to Resyn because the predicted spectrum degraded from the extracted one. Original VC-T 100k scored worse than ConvS2S-VC, but its 1000k-step version (Original VC-T 1000k) outperformed it. This improvement is attributed to VC-T's ability to refine alignments and spectra with training progress, as seen in § 4.4. VC-T pt performed better than ConvS2S-VC or Original VC-T 100k but worse than Original VC-T 1000k. This is because VC-T pt was affected by DTW errors, resulting in unnatural spectra. VC-T ft w/ ali 100k achieved comparable naturalness to Original VC-T 1000k with fewer training steps, revealing the effectiveness of our proposed pre-training. VC-T ft w/o ali 100k significantly outperformed them even without guiding alignments. We argue that this model only took a total of 4.8 days (3.4+1.4 days for 1000k-steps pre-training and 100k-steps fine-tuning), whereas Original VC-T 1000k took 14.4 days (three times faster). Furthermore, VC-T ft w/o ali 1000k achieved the best naturalness among all VC models. This reconfirms that guiding alignment aids training stability but doesn't necessarily yield optimal alignment, as discussed in § 4.4.

Figure 6 (b) presents the speaker similarity scores. The same tendency as naturalness was observed again demonstrating that 1) our pre-training gives better performance and 2) guiding alignment is not necessary in the fine-tuning phase.

5. Conclusion

This paper proposed a novel training pipeline, combining pre-training and fine-tuning, resulting in a faster-converging and higher-performing VC-T without requiring guiding alignment. Applying this approach to streaming VC-T produces commendable WERs and MCDs with significantly fewer training steps. Objective and subjective evaluations showed our finetuned VC-T outperforms models trained from scratch with just one-tenth of the training steps. Notably, our approach enabled VC-T to 1) reduce training time and 2) eliminate need for guiding alignments.

6. References

[1] Yi-Chiao Wu Yu Tsao Chin-Cheng Hsu, Hsin-Te Hwang and HsinMin Wang, "Voice conversion from non-parallel corpora using variational auto-encoder," *Proc. APSIPA*, pp. 1–6, 2016.

[2] Yanping Li, Kong Aik Lee, Yougen Yuan, Haizhou Li, and Zhen Yang, "Many-to-many voice conversion based on bottleneck features with variational autoencoder for non-parallel training data," *Proc. APSIPA*, pp. 829–833, 2018.

[3] Kaizhi Qian, Yang Zhang, Shiyu Chang, Xuesong Yang, and Mark Hasegawa-Johnson, "AutoVC: Zero-shot voice style transfer with only autoencoder loss," *Proc. ICML*, vol. 97, pp. 5210–5219, 2019.

[4] Takuhiro Kaneko and Hirokazu Kameoka, "CycleGAN-VC: Non-parallel voice conversion using cycle-consistent adversarial networks," *Proc. EUSIPCO*, pp. 2100–2104, 2018.

[5] Hirokazu Kameoka, Takuhiro Kaneko, Kou Tanaka, and Nobukatsu Hojo, "StarGAN-VC: Non-parallel many-to-many voice conversion using star generative adversarial networks," *Proc. SLT*, pp. 266–273, 2018.

[6] Bac Nguyen and Fabien Cardinaux, "NVC-Net: End-to-end adversarial voice conversion," *Proc. ICASSP*, pp. 7012–7016, 2022.

[7] "WaveNet: A Generative Model for Raw Audio," Tech. Rep.

[8] Hyeong-Seok Choi, Juheon Lee, Wansoo Kim, Jie Lee, Hoon Heo, and Kyogu Lee, "Neural analysis and synthesis: Reconstructing speech from self-supervised representations," *Proc. NeurIPS*, vol. 34, pp. 16251–16265, 2021.

[9] Wen-Chin Huang, Shu-Wen Yang, Tomoki Hayashi, Hung-Yi Lee, Shinji Watanabe, and Tomoki Toda, "S3PRL-VC: Open-source voice conversion framework with self-supervised speech representations," *Proc. ICASSP*, pp. 6552–6556, 2022.

[10] Wen-Chin Huang, Tomoki Hayashi, Yi-Chiao Wu, Hirokazu Kameoka, and Tomoki Toda, "Pretraining techniques for sequence-to-sequence voice conversion," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 29, pp. 745–755, 2021.

[11] Tomoki Toda, Alan W. Black, and Keiichi Tokuda, "Voice conversion based on maximum likelihood estimation of spectral parameter trajectory," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2222–2235, 2007.

[12] Srinivas Desai, E Veera Raghavendra, B Yegnanarayana, Alan W Black, and Kishore Prahallad, "Voice conversion using artificial neural networks," *Proc. ICASSP*, pp. 3893–3896, 2009.

[13] Kun Li Helen Meng Lifa Sun, Shiyin Kang, "Voice conversion using deep bidirectional long short-term memory based recurrent neural networks," *Proc. ICASSP*, pp. 4869–4873, 2015.

[14] Kazuhiro Kobayashi and Tomoki Toda, "sprocket: Open-source voice conversion software.," *Proc. Odyssey*, pp. 203–210, 2018.

[15] Kou Tanaka, Hirokazu Kameoka, Takuhiro Kaneko, and Nobukatsu Hojo, "AttS2S-VC: Sequence-to-sequence voice conversion with attention and context preservation mechanisms," *Proc. ICASSP*, pp. 6805–6809, 2019.

[16] Hirokazu Kameoka, Kou Tanaka, Damian Kwaśny, Takuhiro Kaneko, and Nobukatsu Hojo, "ConvS2S-VC: Fully convolutional sequence-to-sequence voice conversion," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 28, pp. 1849–1863, 2020.

[17] Wen-Chin Huang, Tomoki Hayashi, Yi-Chiao Wu, Hirokazu Kameoka, and Tomoki Toda, "Voice transformer network: Sequence-to-sequence voice conversion using transformer with text-to-speech pretraining," *Proc. Interspeech*, pp. 4676–4680, 2000.

[18] Hiroki Kanagawa, Takafumi Moriya, and Yusuke Ijima, "VC-T: Stream voice conversion based on neutral transducer," *Proc. Interspeech*, pp. 2088–2092, 2023.

[19] Alex Graves, "Sequence transduction with recurrent neural networks," *Proc. ICML Representation Learning Workshop*, 2012.

[20] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang, "Conformer: Convolution-augmented transformer for speech recognition," *Proc. Interspeech*, pp. 5036–5040, 2020.

[21] Jiawei Chen, Xu Tan, Yichong Leng, Jin Xu, Guihua Wen, Tao Qin, and Tie-Yan Liu, "Speech-T: Transducer for text to speech and beyond," *Proc. NeurIPS*, vol. 34, pp. 6621–6633.

[22] Minchan Kim, Myeonghun Jeong, Byoung Jin Choi, Dongjune Lee, and Nam Soo Kim, "Transduce and speak: Neural transducer for text-to-speech with semantic token prediction," *Proc. ASRU*, pp. 1–7, 2023.

[23] Takafumi Moriya, Takanori Ashihara, Tomohiro Tanaka, Tsubasa Ochiai, Hiroshi Sato, Atsushi Ando, Yusuke Ijima, Ryo Masumura, and Yusuke Shinohara, "SimpleFlat: A simple whole-network pre-training approach for RNN transducer-based end-to-end speech recognition," *Proc. ICASSP*, pp. 5664–5668, 2021.

[24] Vincent Vanhoucke, Matthieu Devin, and Georg Heigold, "Multiframe deep neural networks for acoustic modeling," *Proc. ICASSP*, pp. 7582–7585, 2013.

[25] Heiga Zen, Yannis Agiomyrgiannakis, Niels Egberts, Fergus Henderson, and Przemysław Szczepaniak, "Fast, compact, and high quality LSTM-RNN based statistical parametric speech synthesizers for mobile devices," *Proc. Interspeech*, pp. 2273–2277, 2016.

[26] Yann N. Dauphin, Angela Fan, Michael Auli, and David Grangier, "Language modeling with gated convolutional networks," *Proc. ICML*, vol. 70, pp. 933–941, 2017.

[27] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *Proc. ICLR*, 2015.

[28] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae, "HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis," *Proc. NeurIPS*, vol. 33, pp. 17022–17033, 2020.

[29] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine Mcleavey, and Ilya Sutskever, "Robust speech recognition via large-scale weak supervision," *Proc. ICML*, vol. 202, pp. 28492–28518, 2023.

[30] Arturo Camacho and John Harris, "A sawtooth waveform inspired pitch estimator for speech and music," *The Journal of the Acoustical Society of America*, vol. 124, pp. 1638–52, 2008.