INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Non-parallel Many-to-many Voice Conversion with PSR-StarGAN

Yanping Li[1], Dongxiang Xu[1], Yan Zhang[2], Yang Wang[3], Binbin Chen[3]

[1]College of Telecommunication & Information Engineering, NJUPT, Nanjing, P.R. China
[2]School of Software Engineering, JIT, Nanjing, P.R. China
[3]vivo AI Lab, Shenzhen, P.R. China

liyp@njupt.edu.cn, dongxiangxu@163.com, zy@jit.edu.cn, {yang.wang.rj, bb.chen}@vivo.com

Abstract

Voice Conversion (VC) aims at modifying source speaker's speech to sound like that of target speaker while preserving linguistic information of given speech. StarGAN-VC was recently proposed, which utilizes a variant of Generative Adversarial Networks (GAN) to perform non-parallel many-to-many VC. However, the quality of generated speech is not satisfactory enough. An improved method named "PSR-StarGAN-VC" is proposed in this paper by incorporating three improvements. Firstly, perceptual loss functions are introduced to optimize the generator in StarGAN-VC aiming to learn high-level spectral features. Secondly, considering that Switchable Normalization (SN) could learn different operations in different normalization layers of model, it is introduced to replace Batch Normalization (BN) in StarGAN-VC. Lastly, Residual Network (ResNet) is applied to establish the mapping of different layers between the encoder and decoder of generator aiming to retain more semantic features when converting speech, and to reduce the difficulty of training. Experiment results on the VCC 2018 datasets demonstrate superiority of the proposed method in terms of naturalness and speaker similarity.

Index Terms: voice conversion, StarGAN-VC, perceptual loss, switchable normalization, residual network

1. Introduction

Voice Conversion (VC) is a technique for modifying one's voice to sound like that of another while preserving linguistic information. Recently, considerable effort was spent on the topic of VC. Various tasks can benefit from the advancement of this technique, such as cross-language conversion, movie dubbing, speaking aids, and recovery of impaired speech signal.

Many methods have been applied in VC successfully. Among them, Gaussian Mixture Model (GMM) [1, 2] is one of the most popular methods, which utilizes a statistical parametric model to transform spectral features. Neural network has also been used in VC for its excellent performance, e.g, Deep Neural Network (DNN) [3, 4], Variational Auto Encoder (VAE) [5, 6, 7], cross-domain VAE [8]. Many approaches of VC are categorized as parallel system, which requires accurately aligned parallel source and target utterances. However, more and more attention has been focused on non-parallel VC system, since it is not easy to collect such parallel utterances.

Recently, Generative Adversarial Network (GAN) [9] has been successfully applied to non-parallel VC, which could learn a global generative distribution of the target speech without explicit approximation. There are also some variants in the framework of GAN for VC, such as Variational Autoencoding Wasserstein GAN VC (VAWGAN-VC) [7], Cycle-consistent GAN VC (CycleGAN-VC) [10, 11], non-parallel many-to-many VC with StarGAN (StarGAN-VC) [12]. Among them, VAWGAN-VC directly incorporates a non-parallel VC criterion into the objective function when designing the speech model. However, the quality of generated speech is poor. Previous research has established that CycleGAN-VC provides a breakthrough in methodology, and performs comparably to a parallel VC method without relying on parallel data or time alignment procedure. A large limitation still exists that this approach can only learn one-to-one mapping. In order to overcome these limitations, StarGAN-VC is proposed to learn many-to-many mapping across different attribute domains simultaneously by a generator. Although this method makes multi-domain non-parallel VC techniques get rid of multi-generator, it still needs further improvements.

"PSR-StarGAN-VC" is thus proposed by incorporating Perceptual loss, Switchable Normalization (SN) [13], and Residual Network (ResNet) [14] to StarGAN-VC. Perceptual loss that depends on high-level features from Discriminator, is introduced for better generator's performance. Meanwhile, SN, which learns to select different normalization layers in a DNN in end-to-end manner, is introduced to replace Batch Normalization (BN) [15] in StarGAN-VC for addressing learning-to-normalize problem. In addition, little attention has been paid to the relationship of feature map in model between encoder and decoder in generator in VC tasks. To capture this special relationship and reduce the difficulty of training, ResNet is incorporated by establishing residual connections between the encoder and decoder of generator. Experiment results are carried out on the Voice Conversation Challenge 2018 (VCC 2018) datasets [16]. Subjective evaluation demonstrates that the proposed method outperforms StarGAN-VC in terms of naturalness and speaker similarity.[1]

2. StarGAN-VC

The conventional StarGAN-VC is briefly reviewed in this section.

2.1. Training objectives

Let $x \in \mathrm{R}^{Q \times T_x}$ and $y \in \mathrm{R}^{Q \times T_y}$ be acoustic feature sequences belonging to source speech x and target speech y respectively, where Q denotes the feature dimension. T_x and T_y is the length of spectrum features for x and y, respectively. Attribute label c' and c represents the unique identity of the source and target speakers, respectively.

2.1.1. Adversarial loss

To make a generated spectrum $G(x, c)$ indistinguishable from a target spectrum y, adversarial losses [9] for discriminator D

[1]https://xudongxiang.github.io/PSR-StarGAN-VC.html

http://dx.doi.org/10.21437/Interspeech.2020-1310

and generator G are used respectively:

$$L_{adv}^{D}(D) = -E_{c\sim p(c),y\sim p(y|c)}[\log D(y,c)] - E_{x\sim p(x),c\sim p(c)}[\log(1 - D(G(x,c),c))] \quad (1)$$

$$L_{adv}^{G}(G) = -E_{x\sim p(x),c\sim p(c)}[\log D(G(x,c),c)] \quad (2)$$

where $y\sim p(y|c)$ denotes a training example of an acoustic feature sequence of real speech with attribute c, and $x\sim p(x)$ denotes the acoustic feature with an arbitrary attribute. During training, G generates an acoustic feature $G(x,c)$, which conditioned on both the input feature x and the target domain code c, and attempts to deceive D by minimizing $L_{adv}^{G}(G)$, while D tries to distinguish the real and fake acoustic features by minimizing $L_{adv}^{D}(D)$.

2.1.2. Domain classification loss

To make the generated acoustic feature be similar to that of target speaker, an auxiliary classifier C is used to calculate the domain classification loss. Domain classification losses for classifier C and generator G are defined as follows:

$$L_{cls}^{C}(C) = -E_{c\sim p(c),y\sim p(y|c)}[\log p_C(c|y)] \quad (3)$$

$$L_{cls}^{G}(G) = -E_{x\sim p(x),c\sim p(c)}[\log p_C(c|G(x,c))] \quad (4)$$

where $p_C(c|y)$ is the class probabilities of y and denotes the class probabilities of generated acoustic feature $G(x,c)$. By minimizing the value of $L_{cls}^{C}(C)$, the classifier C is trained for real acoustic features y. Then, the domain classification loss $L_{cls}^{G}(G)$ of $G(x,c)$ is used to optimize G by minimizing $L_{cls}^{G}(G)$ to generate spectral features that can be classified as the target domain c.

2.1.3. Cycle consistency loss

To guarantee linguistic consistency between source spectral features and generated features, cycle consistency loss [10, 17, 18, 19] is introduced as:

$$L_{cyc}(G) = E_{c'\sim p(c),x\sim p(x|c')}[||G(G(x,c),c') - x||_p] \quad (5)$$

where $x\sim p(x|c')$ represents a feature of speech with attribute to c' and denotes p a positive constant. $G(x,c)$ represents the generated features conditioning on x and the target domain c. Meanwhile, $G(G(x,c),c')$ represents the generated features which condition on $G(x,c)$ and c'.

2.1.4. Identity mapping loss

An identity mapping loss

$$L_{id}(G) = E_{c'\sim p(c),x\sim p(x|c')}[||G(x,c') - x||_p] \quad (6)$$

is also used for ensuring that generated spectral features can remain unchanged when the input belongs to the source attribute c'.

2.1.5. Full objective function

On the whole, the target of StarGAN-VC is to minimize these following formulas:

$$L_G(G) = L_{adv}^{G}(G) + \lambda_{cls}L_{cls}^{G}(G) + \lambda_{cyc}L_{cyc}(G) + \lambda_{id}L_{id}(G) \quad (7)$$

$$L_D(D) = L_{adv}^{D}(D) \quad (8)$$

$$L_C(C) = L_{cls}^{C}(C) \quad (9)$$

where λ_{cls}, λ_{cyc} and λ_{id} are regularization parameters that control the relative importance of these losses.

2.2. Network architecture

The network architecture of StarGAN-VC shown in Fig. 3 of [12], consists of generator G, discriminator D, and domain classifier C.

3. PSR-StarGAN-VC

3.1. Motivation

While StarGAN-VC allows non-parallel many-to-many VC, the generated speech still needs further improvements. In view of the many similarities between VC and image style transfer, many techniques can be shared between them. Previous work has shown that high-quality images can be generated by defining and optimizing perceptual loss functions based on high-level features extracted from pretrained networks for image transformation tasks [20, 21]. Considering that perceptual loss could improve details of generated image in conversion tasks, PSR-StarGAN-VC is proposed to optimize StarGAN-VC by utilizing perceptual loss. Generator of the proposed method is trained by inserting perceptual loss to the loss function of generator, not just using per-pixel loss function that depends on low-level pixel information [22]. During training, perceptual loss was calculated by perceptual network, which could extract high-level features from spectrums. In order to address learning-to-normalize problem, SN [13] is introduced to replace BN in StarGAN-VC. Besides, ResNet [14] is also introduced into generator to establish the mapping between the encoder and decoder of generator, and to reduce the difficulty of model training.

3.2. Perceptual loss functions

Perceptual loss consists of two different loss: *content loss* measuring content difference between generated spectral features and source spectral features, and *style loss* measuring style difference between generated spectral features and target spectral features. The latent infromation in low and high dimensions can be extracted by a perceptual network. In the field of image style transfer [22], perceptual network is the 16-layer VGG network [23] that pretrained on the ImageNet dataset [24]. Part of discriminator D is novelly chosen as perceptual network to calculate perceptual loss. Discriminator D, which will be updated continuously during training, can get better representation of our own dataset than other pretrained model, and measure perceptual loss dynamically.

3.2.1. Content loss

The content loss function, which measures the difference of semantic information between generated spectral feature y' and source spectral feature x, is defined as

$$L_{content}{}^{\phi,j}(y',x) = \frac{1}{C_j H_j W_j}||\phi_j(y') - \phi_j(x)||_2^2 \quad (10)$$

where $\phi_j(x)$ is the activations of the j^{th} layer of the perceptual network ϕ when processing x, C_j, H_j and W_j represent the channels, height and width of speech feature map in j^{th} layer of ϕ, respectively. y' are encouraged to be perceptually similar to x, but does not force them to match exactly.

3.2.2. Style loss

The following style loss were proposed in [20, 21]. It starts by defining the gram matrix $G_j^{\phi}(x)_{c,c'}$ to be the matrix $C_j \times C_j$

whose elements are given by

$$G_j^{\phi}(x)_{c,c'} = \frac{1}{C_j H_j W_j} \sum_{h=1}^{H_j} \sum_{w=1}^{W_j} \phi_j(x)_{h,w,c} \phi_j(x)_{h,w,c'} \quad (11)$$

where $\phi_j(x)$ is interpreted as C_j-dimensional features for each point on a $H_j \times W_j$ grid, and $G_j^{\phi}(x)$ is proportional to the uncentered covariance of the C_j-dimensional features. Each grid location is treated as an independent sample. Then, the style loss can be defined to measure the difference between the Gram matrices of the generated spectral features y' and the target spectral features y:

$$L_{style}^{\phi,j}(y', y) = ||G_j^{\phi}(y') - G_j^{\phi}(y)||_2^2 \quad (12)$$

In order to penalize difference in style such as timbre or tone, the 3^{rd} layer of perceptual network ϕ is chosen to extract content representations, and 1^{st}, 2^{nd}, 3^{rd}, and 4^{th} layer of ϕ are all chosen to extract style representation.

$$L_{content} = L_{content}^{\phi,3}(y', x) \quad (13)$$

$$L_{style} = \sum_{i=1}^{4} L_{style}^{\phi,i}(y', y) \quad (14)$$

The total perceptual loss is defined as:

$$L_{perc} = L_{content} + L_{style} \quad (15)$$

Then, inserting L_{perc} to (7) can get an extended loss function:

$$\begin{aligned} L_G(G) = {} & L_{adv}^G(G) + \lambda_{cls} L_{cls}^G(G) \\ & + \lambda_{cyc} L_{cyc}(G) + \lambda_{id} L_{id}(G) + \lambda_{perc} L_{perc} \end{aligned} \quad (16)$$

of generator G. λ_{perc} controls the relative importance of L_{perc} in these losses. By minimizing $L_G(G)$, the generator G can be further optimized that make generated features to be more deceptive. In addition, loss function of discriminator D and classifier C are not changed. Thus, loss functions of D and C in PSR-StarGAN-VC are the same formulas (8) and (9), respectively.

3.3. Switchable normalization

Many normalization methods have been developed, such as B-N [15], Instance Normalization (IN) [25], Layer Normalization (LN) [26] and Group Normalization (GN) [27]. Most of models employed the same normalization technique in all normalization layers of an entire model, and it is expected to achieve sub-optimal performance when specifying normalization method manually. Different normalization methods are suitable to solve different tasks, and the potential of a model's good performance may be impaired when assigning normalization method manually. The generator of StarGAN-VC focuses on extracting information from spectral features, while discriminator and classifier pay attention to extracting features and to classifying extracted features. SN employs three distinct methods (BN, IN and LN) to compute statistics, and switches them by assigning appropriate weights in end-to-end manner. Furthermore, it is verified that SN outperforms its counterparts in various tasks [13]. Thus, BN is replaced with SN in the proposed method. Detailed descriptions of SN are given in [13].

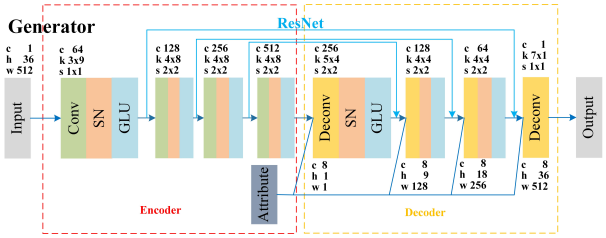

Figure 1: *Network architecture of generator. h, w, and c represent height, width, and number of channels, respectively. In each convolution layer, k, c, and s denote kernel size, number of channels, and stride size, respectively. Conv, SN, GLU, and Deconv denote convolution, switchable normalization, gated linear unit, and transposed convolution.*

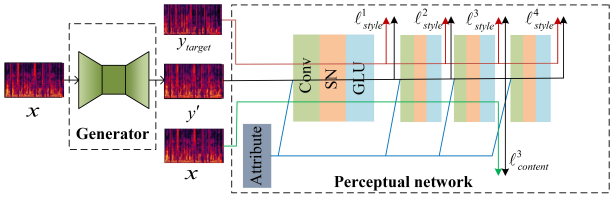

Figure 2: *Perceptual network overview.*

3.4. ResNet

ResNet helps to optimize DNN, to gain benefit from considerably increased depth [14], and to make the model has stronger ability of learning identity mappings which is an appealing property for VC. Many methods in voice conversation utilize ResNet to improve their networks by adding several residual blocks [28, 29]. However, the mapping between the encoder and decoder of generator is still neglected. In order to capture the mapping relationship, ResNet is introduced to the proposed method by establishing residual connection at the same dimension of feature map between the encoder and decoder of generator G. The output of the 1^{st}, 2^{nd} and 3^{rd} layers of the encoder is added to the output of the 3^{rd}, 2^{nd} and 1^{st} layers of the decoder, respectively. The mapping makes the model pay more attention to differences of the source and target speaker's identity in spectrum, and reduces the difficulty of training.

3.5. Network architecture of PSR-StarGAN-VC

PSR-StarGAN-VC consists of a generator G, a real/fake discriminator D, a domain classifier C and a perceptual network calculating perceptual loss. The architecture of the proposed method mostly reuse the architecture of StarGAN-VC in Fig. 3 of [12]. The encoder of G is simplified by cutting it from five layers to four layers and adjusting its parameters of strides, kernels, and channels. The architecture of G is shown in Figure 1. Besides, the feature mapping is established by constructing ResNet between the encoder and decoder of G. SN is also used to maintain reasonably good performance and make model design manageable. Architectures of D and C are not modified except that BN is replaced by SN. Notably, the perceptual network is a part of discriminator, and the first four convolution layers of the discriminator D are utilized as a perceptual network to calculate the perceptual loss novelly shown in Figure 2.

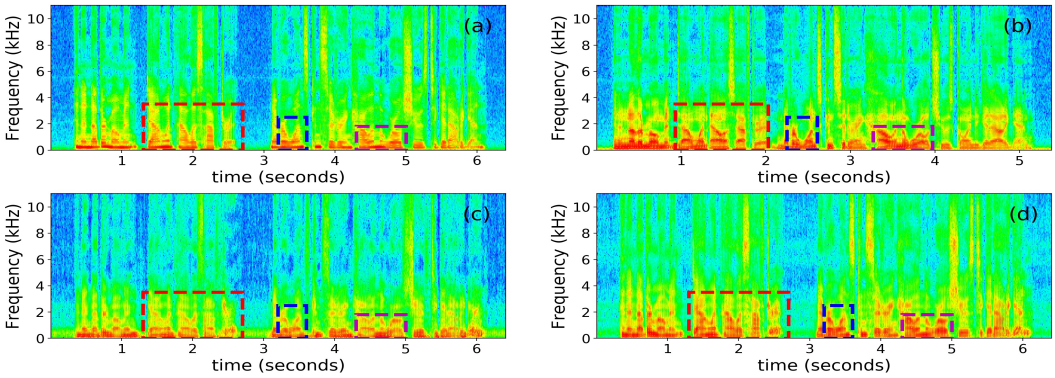

Figure 3: *Example of SF3 to TF1. (a), (b), (c), and (d) are spectrum of source speech, target speech, converted speech obtained with StarGAN-VC, and converted speech obtained with PSR-StarGAN-VC, respectively.*

4. Evaluation

4.1. Experiment conditions

In order to evaluate the proposed method, experiment is carried out on VCC2018 [16], a dataset for VC task. Eight speakers (SF3, SF4, SM3, SM4, TF1, TF2, TM1 and TM2) are selected to perform intra-gender and inter-gender VC. Following the study of StarGAN-VC, for each utterance, a spectral envelope, a logarithmic fundamental frequency ($\log F_0$), and aperiodicities (APs) are extracted every 5ms by using the WORLD vocoder [30]. 36 mel-cepstral coefficients (MCCs) are extracted from spectral envelope in each frame. F_0 is converted by using the logarithm Gaussian normalized transformation [31], and APs are used directly without modification. After preliminary expermiment, λ_{cyc}, λ_{cls}, λ_{id}, and λ_{perc} are fixed to 10, 2 , 5, and 3, respectively. The following two systems are built for comparison:

- Baseline: StarGAN-VC.
- Proposed: PSR-StarGAN-VC.

4.2. Subjective evaluation

To compare naturalness and speaker similarity of generated speech between StarGAN-VC and PSR-StarGAN-VC, naturalness and speaker similarity tests of the generated speech are carried out. Sixteen professional listeners participated in these listening tests. Figure 3 shows an example of spectrum that of source, target, and converted speech with the two systems. It can be seen that PSR-StarGAN-VC outperforms StarGAN-VC in terms of overall styles and local details, such as details of spectrum framed by dotted boxes.

To measure naturalness, Mean Opinion Score (MOS) test is conducted on a 5-point scale (1:bad to 5:excellent). To measure speaker similarity, XAB test is also carried out on the same dataset, where X was target speech, A and B represent generated speech by PSR-StarGAN-VC and StarGAN-VC. For each sentence pair, listeners need to classify speech to the most suitable choice (A, B, or Fair). As shown in Figure 4 and 5, the proposed method achieved improved performance than the baseline method in all test pairs.

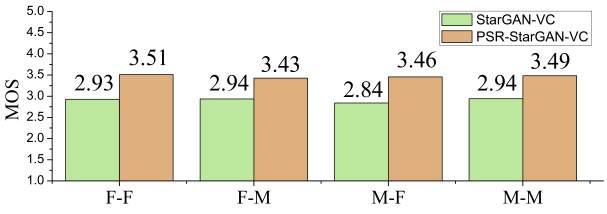

Figure 4: *MOS for naturalness comparison.*

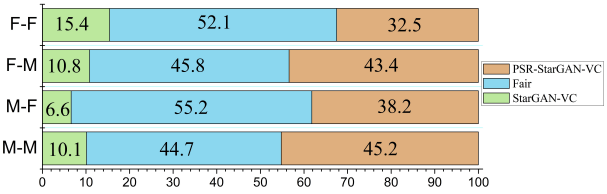

Figure 5: *Average preference score (%) on speaker similarity.*

5. Conclusions

A novel method named "PSR-StarGAN-VC" is designed by incorporating three improvements to StarGAN-VC. Firstly, perceptual loss functions based on high-level spectrum features extracted from perceptual network is defined to generate high-quality speech. Secondly, learning-to-normalize problem is addressed by assigning SN to select proper normalizers for different normalization layers of the entire model. Lastly, ResNet is constructed in generator to encourage the generator to focus on capturing style's difference between source speech and target speech by creating short connections during encoder and decoder of generator. Experiment results verified that PSR-StarGAN-VC outperforms StarGAN-VC in both naturalness and speaker similarity. During writing this paper, StarGAN-VC2 [32] is proposed and achieves good performance. Applying these improvements above to StarGAN-VC2 will be an interesting direction in future.

6. Acknowledgements

This work is supported by the National Nature Science Foundation of China under Grant No. 61401227, No. 61872199, No. 61872424, Special Project in Jinling Institute of Technology for Building Innovative Team on Intelligent Human Computer Interaction (218/010119200113).

7. References

[1] Y. Stylianou, O. Cappé, and E. Moulines, "Continuous Probabilistic Transform for Voice Conversion," *IEEE Transactions on Speech and Audio Processing*, vol. 6, no. 2, pp. 131–142, 1998.

[2] E. Helander, T. Virtanen, J. Nurminen, and M. Gabbouj, "Voice Conversion Using Partial Least Squares Regression," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 5, pp. 912–921, 2010.

[3] F.-L. Xie, F. K. Soong, and H. Li, "A KL Divergence and DNN-Based Approach to Voice Conversion without Parallel Training Sentences," in *Interspeech*, 2016, pp. 287–291.

[4] L.-H. Chen, Z.-H. Ling, L.-J. Liu, and L.-R. Dai, "Voice Conversion Using Deep Neural Networks with Layer-Wise Generative Training," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 12, pp. 1859–1872, 2014.

[5] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H.-M. Wang, "Voice Conversion from Non-Parallel Corpora Using Variational Auto-Encoder," in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*. IEEE, 2016, pp. 1–6.

[6] Y. Saito, Y. Ijima, K. Nishida, and S. Takamichi, "Non-Parallel Voice Conversion Using Variational Autoencoders Conditioned by Phonetic Posteriorgrams and D-vectors," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5274–5278.

[7] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H.-M. Wang, "Voice Conversion From Unaligned Corpora Using Variational Autoencoding Wasserstein Generative Adversarial Networks," *arXiv preprint arXiv:1704.00849*, 2017.

[8] W.-C. Huang, H.-T. Hwang, Y.-H. Peng, Y. Tsao, and H.-M. Wang, "Voice Conversion Based on Cross-Domain Features Using Variational Auto Encoders," in *2018 11th International Symposium on Chinese Spoken Language Processing (ISCSLP)*. IEEE, 2018, pp. 51–55.

[9] T. Kaneko, H. Kameoka, K. Hiramatsu, and K. Kashino, "Sequence-to-Sequence Voice Conversion with Similarity Metric Larned Using Generative Adversarial Networks," in *INTERSPEECH*, vol. 2017, 2017, pp. 1283–1287.

[10] T. Kaneko and H. Kameoka, "Parallel-Data-Free Voice Conversion Using Cycle-Consistent Adversarial Networks," *arXiv preprint arXiv:1711.11293*, 2017.

[11] F. Fang, J. Yamagishi, I. Echizen, and J. Lorenzo-Trueba, "High-Quality Nonparallel Voice Conversion Based on Cycle-Consistent Adversarial Network," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5279–5283.

[12] H. Kameoka, T. Kaneko, K. Tanaka, and N. Hojo, "StarGAN-VC: Non-Parallel Many-to-Many Voice Conversion Using Star Generative Adversarial Networks," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 266–273.

[13] P. Luo, R. Zhang, J. Ren, Z. Peng, and J. Li, "Switchable Normalization for Learning-to-Normalize Deep Representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[15] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," *arXiv preprint arXiv:1502.03167*, 2015.

[16] J. Lorenzo-Trueba, J. Yamagishi, T. Toda, D. Saito, F. Villavicencio, T. Kinnunen, and Z. Ling, "The Voice Conversion Challenge 2018: Promoting Development of Parallel and Nonparallel Methods," *arXiv preprint arXiv:1804.04262*, 2018.

[17] T. Zhou, P. Krahenbuhl, M. Aubry, Q. Huang, and A. A. Efros, "Learning Dense Correspondence via 3d-guided Cycle Consistency," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 117–126.

[18] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2223–2232.

[19] J. Li, E. Chen, Z. Ding, L. Zhu, K. Lu, and Z. Huang, "Cycle-Consistent Conditional Adversarial Transfer Networks," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 747–755.

[20] L. Gatys, A. S. Ecker, and M. Bethge, "Texture Synthesis Using Convolutional Neural Networks," in *Advances in Neural Information Processing systems*, 2015, pp. 262–270.

[21] L. A. Gatys, A. S. Ecker, and M. Bethge, "A Neural Algorithm of Artistic Style," *arXiv preprint arXiv:1508.06576*, 2015.

[22] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual Losses for Real-Time Style Transfer and Super-Resolution," in *European Conference on Computer Vision*. Springer, 2016, pp. 694–711.

[23] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[24] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[25] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance Normalization: The Missing Ingredient for Fast Stylization," *arXiv preprint arXiv:1607.08022*, 2016.

[26] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer Normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[27] Y. Wu and K. He, "Group Normalization," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 3–19.

[28] T. Kaneko, H. Kameoka, K. Tanaka, and N. Hojo, "CycleGAN-VC2: Improved Cyclegan-Based Non-Parallel Voice Conversion," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6820–6824.

[29] H. Ren, M. El-Khamy, and J. Lee, "Dn-Resnet: Efficient Deep Residual Network for Image Denoising," in *Asian Conference on Computer Vision*. Springer, 2018, pp. 215–230.

[30] M. Morise, F. Yokomori, and K. Ozawa, "World: A Vocoder-Based High-Quality Speech Synthesis System for Real-Time Applications," *IEICE TRANSACTIONS on Information and Systems*, vol. 99, no. 7, pp. 1877–1884, 2016.

[31] K. Liu, J. Zhang, and Y. Yan, "High Quality Voice Conversion through Phoneme-Based Linear Mapping Functions with Straight for Mandarin," in *Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007)*, vol. 4. IEEE, 2007, pp. 410–414.

[32] T. Kaneko, H. Kameoka, K. Tanaka, and N. Hojo, "StarGAN-VC2: Rethinking Conditional Methods for StarGAN-Based Voice Conversion," *arXiv preprint arXiv:1907.12279*, 2019.