INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

Generative Adversarial Networks for Unpaired Voice Transformation on Impaired Speech

Li-Wei Chen[1], *Hung-Yi Lee*[1], *Yu Tsao*[2]

[1]National Taiwan University, Taiwan
[2]Academia Sinica, Taiwan

b04901014@ntu.edu.tw, hungyilee@ntu.edu.tw, yu.tsao@citi.sinica.edu.tw

Abstract

This paper focuses on using voice conversion (VC) to improve the speech intelligibility of surgical patients who have had parts of their articulators removed. Due to the difficulty of data collection, VC without parallel data is highly desired. Although techniques for unparallel VC—for example, CycleGAN—have been developed, they usually focus on transforming the speaker identity, and directly transforming the speech of one speaker to that of another speaker and as such do not address the task here. In this paper, we propose a new approach for unparallel VC. The proposed approach transforms impaired speech to normal speech while preserving the linguistic content and speaker characteristics. To our knowledge, this is the first end-to-end GAN-based unsupervised VC model applied to impaired speech. The experimental results show that the proposed approach outperforms CycleGAN.

Index Terms: Unpaired Voice Transformation, Generative Adversarial Networks

1. Introduction

Voice conversion (VC) is a task aimed at converting the speech signals from a certain acoustic domain to another while keeping the linguistic content the same. Examples of acoustic domains include not only speaker identity [1, 2, 3, 4], but many other factors orthogonal to the linguistic content, such as speaking style, speaking rate [5], noise condition, emotion [6, 7], and accent [8], with potential applications ranging from speech enhancement [9, 10], computer-assisted pronunciation training for non-native language learner [8], speaking assistance [11], to name a few.

This paper focuses on using VC to improve the speech intelligibility of surgical patients who have had parts of their articulators removed. Because of the removal of parts of the articulator, a patient's speech may become distorted and difcult to understand. VC methods can be applied to convert the distorted speech such that it is clear and more intelligible. In this work, we consider a VC model without ASR [12] because collecting a large amount of data to train an ASR for impaired speech is laborious and not practical. Non-negative matrix factorization (NMF) based VC has been used for this task [13, 14, 15]. In previous work, paired utterances from both patients and unimpaired people were needed for training. Collecting a large amount of audio from patients is difficult under this task because even speaking for a long time is usually difficult for them, not to mention the collection of paired data. Due to the lack of training data, to our best knowledge, deep learning has not been widely applied on this task yet.

After the success of deep learning in various domains, many researchers have attempted to incorporate deep learning into the VC framework, but most focus on speaker identity conversion. Most previous work requires aligned data, but due to the difficulties in obtaining aligned data, approaches utilizing generative models such as variational autoencoders (VAEs) [16, 17] and generative adversarial networks (GANs) [18, 19] were studied because they can be trained with non-parallel data. VC for articulation disorders without parallel data is highly desired due to the difficulty of data collection. To achieve that, one can simply apply the techniques developed for speaker identity VC by considering the patient with the articulation disorder as the source speaker, and the unimpaired person as the target speaker. However, the model thus learned would simply convert the voice of the source speaker into that of the target speakers without preserving the source speaker's individuality. Even worse, the speaker VC model may change only speaker characteristics, but yield a converted voice that is still unclear. Therefore, to achieve VC for articulation disorders without parallel data, a new approach must be developed.

The overview of the proposed approach is shown in Figure 1. The proposed model includes a generator, a discriminator, and a controller. The generator and discriminator form a GAN which is learned from a large amount of normal speech which is easier to collect than impaired speech. The discriminator learns to judge whether the input is real speech or if it has been generated by the generator. The generator takes a code which represents the content and speaker of the audio to be generated as input, and generates normal speech to fool the discriminator. The impaired speech is used only to train the controller. Given impaired speech as input, the controller outputs a code which is taken as the input of the generator, and the generator generates normal speech based on the input code. The controller learns to generate code that makes the generator output normal speech with the same linguistic content and the same speaker characteristics as the impaired speech, thus minimizing their high-level differences. To guide the controller learning, we require automatic ways to evaluate this high-level difference. Inspired by perception loss, widely used in image processing [20], we use the hidden layers of the discriminator to evaluate the similarity of two audio segments. Compared with CycleGAN, which maps from the source speaker to the target speaker also in an unsupervised way, the proposed approach better improves speech intelligibility while preserving speaker characteristics.

2. Proposed Approaches

The proposed approach consists of three models: a generator G, a discriminator D, and a controller C. For training data, we have a large amount of speech from unimpaired subjects: $\mathcal{T} = \{x_i^t\}_{i=1}^N$, where x_i^t is a fixed-length acoustic feature sequence from the utterances of unimpaired subjects, and N is the number of audio segments in the training set. We also have the speech of a patient, $\mathcal{S} = \{x_i^s\}_{i=1}^{N'}$, where N' is the number of audio segments for the patient, but the data size is much smaller than that of unimpaired subjects ($N' \ll N$). The content of normal speech and impaired speech are completely unrelated. During testing, given an utterance of the patient, it is first equally segmented into a sequence of audio segments. The controller takes the audio segments as input,

http://dx.doi.org/10.21437/Interspeech.2019-1265

and the generator transforms them into normal speech.

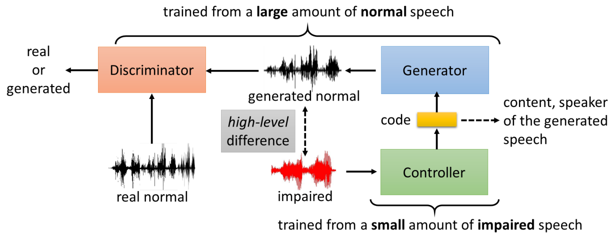


Figure 1: *Overview of proposed approach. The difference is evaluated by a network instead of using low-level signal differences. The controller learns to minimize the high-level difference.*

2.1. Generator-Discriminator

We use the audio of unimpaired subjects $\mathcal{T}$ to train the audio generator G and discriminator D. The generator is used to generate audio $\tilde{x}$ given a vector c, that is, $\tilde{x} = G(c)$, and the discriminator D attempts to distinguish $x^t \sim \mathcal{T}$ from $\tilde{x} \sim G$ while the generator tries to fool it. As shown below, the objective functions for D and G follow the idea of LSGAN [21]:

$$\begin{aligned} \mathcal{L}_D &= \mathbb{E}_{x^t \sim \mathcal{T}}[(D(x^t) - 1)^2] \\ &\quad + \mathbb{E}_{c \sim P_c(c), \tilde{x} \sim G(c)}[(D(\tilde{x}))^2] \end{aligned} \tag{1}$$

$$\mathcal{L}_G = \mathbb{E}_{c \sim P_c(c), \tilde{x} \sim G(c)}[(D(\tilde{x}) - 1)^2] \tag{2}$$

In (1), D learns to assign normal speech x^t a score of one, and assign a score of zero to generated audio segments $\tilde{x}$. At the same time, in (2), G learns to generate an $\tilde{x}$ that yields a score of one from D. Here c is the output of the controller C, which we assume has a distribution $P_c(c)$ in (1) and (2)[1]. After the above training procedure, we have a generator G which generates normal speech given a condition vector c. The vector c controls the generated audio of G. By choosing the condition vector c properly, we generate audio segments with the desired content and speaker characteristic. The core idea is that a large amount of normal speech can be used to train a generator G which can generate high quality speech, and the impaired speech is only used to select c, which is a much easier task than speech generation.

2.2. Controller

Given the audio segment x^s from a patient, we want to find its corresponding counterpart x^t in the domain of normal speech. The basic idea is to properly choose the condition c that causes G to generate speech x^t similar to x^s. If x^t is close to x^s, they may contain the same linguistic information with the same speaker characteristics, but the x^t generated by G sounds like normal speech (which is what we want) because G has learned to generate normal speech.

The controller C takes an audio segment x^s as input, and outputs its corresponding condition c as the input of G. Here we assume only a small amount of audio $\mathcal{S}$ from the patient is available as training data. $\mathcal{S}$ is used only to train controller C. C is learned by minimizing the following loss:

$$\mathcal{L}_C = \mathbb{E}_{x^s \sim \mathcal{S}}[L(G(C(x^s)), x^s)] \tag{3}$$

The metric $L(.,.)$ is used to evaluate the difference between two audio segments. In (3), C learns to make the input x^s and the corresponding output of the generator $G(C(x^s))$ as close as possible. $L(.,.)$ is defined in the next subsection. If we jointly optimize G and C, minimizing (3) is equivalent to training an auto-encoder (the controller is an encoder, while the generator is a decoder). However, we only update C when we minimize (3). This is very critical for the success of this approach, because if G is also updated to minimize (3), we cannot guarantee that G still generates normal speech after the update.

2.3. Distance Measure for Audio

For distance $L(.,.)$, both L1 and L2 loss are not suitable because we seek to evaluate the similarity of the content and the speaker characteristics between two audio segments, not merely low-level signal similarity. On image style transfer tasks, the perceptual loss [20], which utilizes the layers of a CNN classifier as features and applies a distance measure to these, has been shown to produce finer results than pixel-wise loss. Here we borrow this idea to evaluate high-level audio similarity. Instead of training another classifier, we use the discriminator as the objective classifier for the distance measure.

For the perceptual loss, we choose the Laplacian pyramid Lap$_1$ loss [22]. We use the notation $D_l(x)$ to denote the output of the l-th layer of the discriminator D given input x. Then $L(.,.)$ in (3) is formulated as

$$L(x, x') = \sum_l 2^{-2l} |D_l(x) - D_l(x')|_1 \tag{4}$$

The L1 distance of the hidden layer output $D_l(x)$ is computed. In (4), all the hidden layers of D are considered to capture information at different granularities. The weights for each layer follows [22].

3. Implementation

3.1. Acoustic Feature Processing

We use the mel spectrogram as the input of the controller and the output of the generator. Before transforming into the spectrogram, we trim audio silence and perform volume normalization. All audio is converted to a 16kHz sample rate. After that, we use a 50 milliseconds window length, a 12.5 milliseconds hop length, and a 1024 FFT window size for the STFT.

After constructing the spectrogram, we construct the mel spectrogram using 128 mel frequency bands with a frequency range from 55Hz to 7600Hz. Then we turn the mel spectrogram into the decibel scale and standardize the features across the time dimension to zero mean and unit variance. We clip the values between $-c$ and c. Although the hyperparameter c is somewhat data dependent, we find that $c = 3$ works well in most cases. Using these settings, most of the human speech mel spectrograms can be transformed back to the raw waveforms with little audible distortion. Since the feature values are constrained to $[-c, c]$, we use $c \cdot tanh(\cdot)$ as the output activation for our generator.

To convert the mel spectrogram back to the raw waveform, we first rescale the model output and then multiply the psuedo inverse of the mel filter bank to recover the linear spectrogram. Finally, using Griffin and Lim's algorithm to estimate the phase, we reconstruct the raw waveform.

[1] To be specific, $c \sim P_c(c)$ is equivalent to $c \sim C(x^s), x^s \sim \mathcal{S}$. This is made clear below.

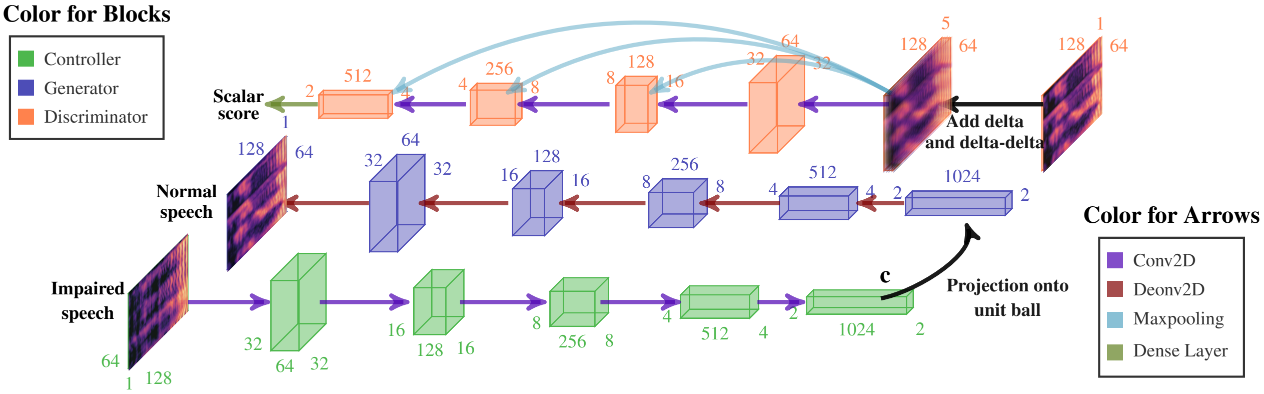


Figure 2: *Controller-generator-discriminator configuration*

3.2. Model Configuration

The network structure of the discriminator, generator, and controller is shown in Figure 2. So that the discriminator better captures dynamics, we augment the input spectrogram with the first and second order deltas in both time and frequency dimensions. The time-frequency bin of the original spectrum is represented by a scalar; after augmentation, each bin is a 5-dimensional vector. To make gradient propagation more efficient, we augment each hidden layer of the discriminator with its input. However, since the discriminator uses strided convolutions, the spacial dimension of the input and hidden layers will be inconsistent. To overcomes this, we apply max pooling on the input to make the spacial dimension consistent with hidden layers, then augment them as additional channels to the hidden layers.

The pool size and stride are determined by the shape of the corresponding discriminator layer. Before c is fed into the generator, we project c onto the L_2 unit ball as

$$c \leftarrow \frac{c}{max(\|c\|_2, 1)} \tag{5}$$

In the real-world implementation, $\mathcal{L}_D$, $\mathcal{L}_G$, and $\mathcal{L}_C$ are updated once in each training iteration.

3.3. Training Details

We trained the proposed model using the Adam optimizer with a 0.0002 learning rate, and $\beta_1 = 0.5$ and $\beta_2 = 0.9$ for the controller and generator. A learning rate of 0.0001 was applied to the discriminator to avoid fast convergence. Instance normalization [23] and spectral normalization [24] were applied to both the generator and discriminator to stabilize training. ELU activation [25] was used for the generator and controller, and RELU activation was used for the discriminator. We used a batch size of 64, a dropout rate of 0.9 for the controller, and 0.8 for the discriminator. Further details may be found in our implementation code: `https://github.com/b04901014/ISGAN`.

4. Experiments

4.1. Experimental Setup

The normal utterances for training the generator and discriminator came from the ASTMIC dataset, which contains 100 male speakers with different speaking rates, each reading out 200 different Mandarin sentences. The impaired speech, from an orally-impaired male speaker, included 132 utterances in Mandarin. As this speaker had a serious articulatory injury, most of his utterances were unintelligible to normal people.

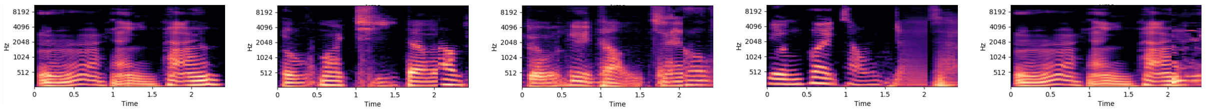
We compare the proposed approach with two baselines:

- Conditional GAN (cGAN): As this method is supervised, parallel audio is needed. A normal male speaker reads the same sentences as the patient, and the paired data is obtained by aligning the utterances of the patient and the normal speaker. cGAN involves training a discriminator and a conditional generator which takes impaired speech as input and outputs normal speech. Here the cGAN training algorithm is that from Pix2pix [26]. The network architecture of the discriminator in cGAN is the same as in the proposed approach, and the network architecture of the conditional generator is the same as the cascaded controller and generator in the proposed approach.
- CycleGAN [27]: As with the proposed approach, this method needs no parallel data. With data from two domains, CycleGAN learns a generator to transform an object from one domain to another domain. Here we consider the ASTMIC audio as one domain and the audio of the patient as the other domain. The discriminator used in CycleGAN also has the same network structure as the proposed approach. Likewise, the generator in CycleGAN is the concatenation of the controller and the generator in the proposed approach.

4.2. Subjective Evaluation

To determine whether these models improve intelligibility while preserving the content and speaker identity, we evaluate the MOS (mean opinion score) on CycleGAN, cGAN, and our model. Figure 4 shows our MOS results. Given the original utterance and the random shuffled utterances transformed by different models, five subjects are asked to evaluate the audio from three aspects: (1) how similar are the speaker characteristics before and after transformation (similarity-speaker); (2) how similar is the linguistic content before and after transformation (similarity-content); and (3) how clear is it compared to normal speech (articulation). The percentiles of the MOS are normalized scores indicating the extent of which subjects consider samples to be similar or articulate respectively.

The similarity MOS indicates that our model does better than cGAN and CycleGAN in preserving both speaker characteristics and linguistic information. CGAN performs the worst despite the additional use of ground truth information, because the



(a) *Impaired speech* (b) *Proposed model* (c) *NoSD model* (d) *CycleGAN transformed* (e) *CycleGAN reconstructed*

Figure 3: *Spectrogram of impaired speech before and after transformation by each model. (a) impaired speech, (b) transformed speech using proposed method, (c) discriminator without skip connection (NoSD), (d) transformed speech using CycleGAN, and (e) reconstructed speech using CycleGAN.*

amount of paired data is not sufficient to train the network. In Section 4.5 we further analyze why CycleGAN does not preserve the speaker characteristics and linguistic information. The articulation MOS shows our improvement in intelligibility over impaired speech. Audio samples for different approaches may be accessed at `https://b04901014.github.io/ISGAN/`.

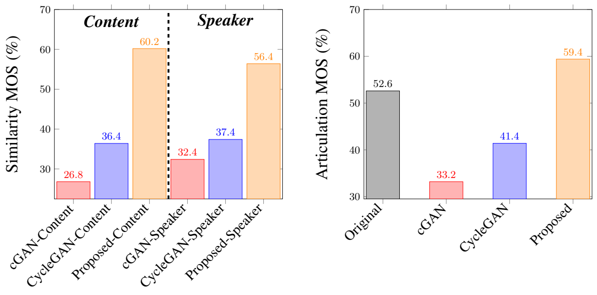
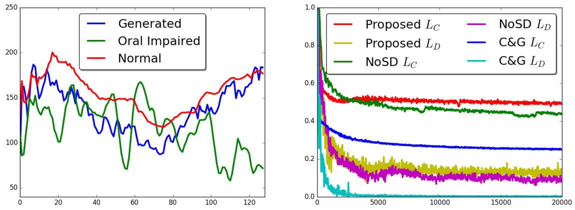


Figure 4: *Mean opinion score (MOS) comparison*



(a) *Global variance* (b) *Loss curve*

Figure 5: *(a) GV before/after transformation by the proposed approach and normal utterances. (b) Loss curve of the models for the ablation study described in Section 4.4.*

4.3. Analysis of Proposed Approach

We first show the global variance (GV) for the proposed model. As shown in Figure 5a, the GV of impaired speech is quite different from that of normal speech. The GV of the generated speech is similar to that of normal utterances. Figures 3a and 3b show an example of an impaired utterance and its transformed results using the proposed approach. As shown in Figure 3a, the orally impaired subject tends to have vague word boundaries, causing the continuous forms on the low frequency bands of the spectrogram. Figure 3b shows the ability to separate entangled word boundaries for the first and second word. This can also be heard in the audio samples. Nevertheless, we also see an obvious artifact around 2s in Figure 3b. This discontinuity is a consequence of our feeding the model of each time window independently, without any information from the previous window. This may be solved in future work by feeding the previous window to the model as an augmented condition.

4.4. Ablation Study

To show the contribution of each part of our model to training stability and audio quality, we conducted an ablation study. We studied three different models: (i) the proposed model, (ii) the model without augmented input to the discriminator (NoSD)[2], and (iii) the model in which the parameters of the generator G and controller C are updated to minimize both Equation (5) and (2). That is, G and C are trained jointly without separate objectives (C&G). Figures 5b and 3c show the functionality of the augmented inputs of the discriminator. The NoSD model gets both lower controller loss ($\mathcal{L}_C$) and discriminator loss ($\mathcal{L}_D$). This indicates the controller is more capable of deceiving the generator, and as a consequence, the generator has less ability to generate plausible results to confuse the discriminator. Thus the NoSD model yields a blurrier spectrogram in Fig. 3c than that for the proposed model in Fig. 3b. When updating G and C jointly (C&G), Figure 5b shows that the discriminator loss $\mathcal{L}_D$ quickly goes to zero, that is, the discriminator easily separates the real normal speech and generator output. This indicates that the generator output can no longer be similar to the normal speech.

4.5. Steganography of CycleGAN

As mentioned in [28], CycleGAN learns to hide the information needed for reconstruction from the source domain into the target domain in an imperceptible manner. We also see this phenomenon in Figures 3d and 3e. The spectrograms before and after the CycleGAN transformation are quite different (Figure 3a vs Figure 3d), whereas after transforming back to the source domain, the reconstructed audio is almost the same as the original input (Figure 3a vs Figure 3e). This indicates that the cycle-consistency loss is not a good regularizer to enforce the model to have consistent input-output pairs. Instead of using cycle-consistency loss, our method utilizes Equation (4) to maintain the consistency of content and speaker identity of the impaired and generated audio. As shown in Figure 4, the proposed approach is better.

5. Concluding Remarks

Here we propose a novel unparallel VC model to improve the speech intelligibility of surgical patients who have had parts of their articulators removed. In comparison with CycleGAN, which also needs only unparallel data, the proposed approach not only better improves articulation but also better preserves the linguistic content and speaker characteristics.

[2] In Figure 2, the light blue arrows in the discriminator are removed.

6. 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] A. Kain and M. W. Macon, "Spectral voice conversion for text-to-speech synthesis," in *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, vol. 1. IEEE, 1998, pp. 285–288.

[3] D. Saito, K. Yamamoto, N. Minematsu, and K. Hirose, "One-to-many voice conversion based on tensor representation of speaker space," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[4] T. Kinnunen, L. Juvela, P. Alku, and J. Yamagishi, "Non-parallel voice conversion using i-vector plda: Towards unifying speaker verification and transformation," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 5535–5539.

[5] D. Rentzos, S. Vaseghi, E. Turajlic, Q. Yan, and C.-H. Ho, "Transformation of speaker characteristics for voice conversion," in *Automatic Speech Recognition and Understanding, 2003. ASRU'03. 2003 IEEE Workshop on*. IEEE, 2003, pp. 706–711.

[6] R. Aihara, R. Takashima, T. Takiguchi, and Y. Ariki, "GMM-based emotional voice conversion using spectrum and prosody features," *American Journal of Signal Processing*, vol. 2, no. 5, pp. 134–138, 2012.

[7] H. Kawanami, Y. Iwami, T. Toda, H. Saruwatari, and K. Shikano, "GMM-based voice conversion applied to emotional speech synthesis," in *Eighth European Conference on Speech Communication and Technology*, 2003.

[8] K. Oyamada, H. Kameoka, T. Kaneko, H. Ando, K. Hiramatsu, and K. Kashino, "Non-native speech conversion with consistency-aware recursive network and generative adversarial network," in *Proceedings of APSIPA Annual Summit and Conference*, vol. 2017, 2017, pp. 12–15.

[9] A. B. Kain, J. P. Hosom, X. Niu, J. P. van Santen, M. Fried-Oken, and J. Staehely, "Improving the intelligibility of dysarthric speech," *Speech communication*, vol. 49, no. 9, pp. 743–759, 2007.

[10] F. Rudzicz, "Acoustic transformations to improve the intelligibility of dysarthric speech," in *Proceedings of the Second Workshop on Speech and Language Processing for Assistive Technologies*. Association for Computational Linguistics, 2011, pp. 11–21.

[11] K. Nakamura, T. Toda, H. Saruwatari, and K. Shikano, "Speaking-aid systems using gmm-based voice conversion for electrolaryngeal speech," *Speech Communication*, vol. 54, no. 1, pp. 134–146, 2012.

[12] E. Dunbar, R. Algayres, J. Karadayi, M. Bernard, J. Benjumea, X.-N. Cao, L. Miskic, C. Dugrain, L. Ondel, A. W. Black, L. Besacier, S. Sakti, and E. Dupoux, "The Zero Resource Speech Challenge 2019: TTS without T," in *20th Annual Conference of the International Speech Communication Association (INTERSPEECH 2019): Crossroads of Speech and Language*, 2019, submitted. [Online]. Available: https://zerospeech.com/2019/

[13] S. Fu, P. Li, Y. Lai, C. Yang, L. Hsieh, and Y. Tsao, "Joint dictionary learning-based non-negative matrix factorization for voice conversion to improve speech intelligibility after oral surgery," *IEEE Transactions on Biomedical Engineering*, vol. 64, 2017.

[14] R. Aihara, R. Takashima, T. Takiguchi, and Y. Ariki, "Consonant enhancement for articulation disorders based on non-negative matrix factorization," in *Proceedings of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, 2012, pp. 1–4.

[15] ——, "Individuality-preserving voice conversion for articulation disorders based on non-negative matrix factorization," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 8037–8040.

[16] C. Hsu, H. Hwang, Y. Wu, Y. Tsao, and H. Wang, "Voice conversion from non-parallel corpora using variational auto-encoder," in *Proceedings of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, 2016, pp. 1–6.

[17] ——, "Voice conversion from unaligned corpora using variational autoencoding wasserstein generative adversarial networks," in *Proceedings of the Interspeech 2017, 18th Annual Conference of the International Speech Communication Association*, 2017, pp. 3364–3368.

[18] Y. Gao, R. Singh, and B. Raj, "Voice impersonation using generative adversarial networks," in *Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 2506–2510.

[19] J. Chou, C. Yeh, H. Lee, and L. Lee, "Multi-target voice conversion without parallel data by adversarially learning disentangled audio representations," in *Proceedings of the Interspeech 2018, 19th Annual Conference of the International Speech Communication Association*, 2018, pp. 501–505.

[20] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proceedings of the Computer Vision - ECCV 2016 - 14th European Conference, Part II*, 2016, pp. 694–711.

[21] X. Mao, Q. Li, H. Xie, R. Lau, Z. Wang, and S. Smolley, "Least squares generative adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2813–2821.

[22] P. Bojanowski, A. Joulin, D. Lopez-Paz, and A. Szlam, "Optimizing the latent space of generative networks," in *Proceedings of the 35th International Conference on Machine Learning (ICML)*, 2018, pp. 599–608.

[23] D. Ulyanov, A. Vedaldi, and V. S. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *CoRR*, vol. abs/1607.08022, 2016.

[24] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," *CoRR*, vol. abs/1802.05957, 2018.

[25] D. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2016.

[26] P. Isola, J. Zhu, T. Zhou, and A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5967–5976.

[27] J. Zhu, T. Park, P. Isola, and A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2242–2251.

[28] C. Chu, A. Zhmoginov, and M. Sandler, "Cyclegan, a master of steganography," *CoRR*, vol. abs/1712.02950, 2017.