Novel Adaptive Generative Adversarial Network for Voice Conversion

Maitreya Patel[1], Mihir Parmar[2], Savan Doshi[1], Nirmesh J. Shah[1] and Hemant A. Patil[1]
[1]Speech Research Lab, Dhirubhai Ambani Institute of Information and Communication Technology, Gandhinagar, India.
[2]Arizona State University, Tempe, USA.
{maitreya_patel, savan_doshi, nirmesh88_shah, hemant_patil}@daiict.ac.in, mparmar3@email.asu.edu

***Abstract*—Voice Conversion (VC) converts the speaking style of a source speaker to the speaking style of a target speaker by preserving the linguistic content of a given speech utterance. Recently, Cycle Consistent Adversarial Network (CycleGAN), and its variants have become popular for non-parallel VC tasks. However, CycleGAN uses two different generators and discriminators. In this paper, we introduce a novel Adaptive Generative Adversarial Network (AdaGAN) for non-parallel VC task, which effectively requires single generator, and two discriminators for transferring the style from one speaker to another while preserving the linguistic content in the converted voices. To the best of authors' knowledge, this is the first study of its kind to introduce a new Generative Adversarial Network (GAN)-based architecture (i.e., AdaGAN) in machine learning literature, and the first attempt to apply this architecture for non-parallel VC task. In this paper, we compared the results of the AdaGAN w.r.t. state-of-the-art CycleGAN architecture. Detailed subjective and objective tests are carried out on the publicly available VC Challenge 2018 corpus. In addition, we perform three statistical analysis which show effectiveness of AdaGAN over CycleGAN for parallel-data free one-to-one VC. For inter-gender and intra-gender VC, We observe that the AdaGAN yield objective results that are comparable to the CycleGAN, and are superior in terms of subjective evaluation. A subjective evaluation shows that AdaGAN outperforms CycleGAN-VC in terms of naturalness, sound quality, and speaker similarity. AdaGAN was preferred 58.33% and 41% time more over CycleGAN in terms of speaker similarity and sound quality, respectively.**

***Index Terms*—Voice Conversion, GAN, CycleGAN, AdaIN, AdaGAN.**

I. Introduction

Voice Conversion (VC) is a technique for converting the voice of source speaker into target speaker by modifying *prosodic* and *spectral* features of the source speaker [1]. Some of the important details in the converted voice are lost because of inaccurate spectral feature mapping and statistical averaging (i.e., over smoothing) of speech sound units [2]. Hence, this problem can be formulated as a regression problem for estimating mapping between para/non-linguistic information that is contained in a given utterance while preserving linguistic content as it is.

Many successful frameworks for VC are based on Gaussian Mixture Models (GMMs) [2], [3], Deep Neural Network (DNN) [4], [5], Recurrent Neural Networks (RNNs) [6], Generative Adversarial Networks (GANs) [7], and Non-negative Matrix Factorization (NMF) [8]. These methods get impressive results in the field of VC using parallel corpus however, among these methods, many VC methods need time alignment between source and target speakers' training data. Even though we could collect such data, manually aligning them can be intensively laborious and costly. In addition, misalignment leads to degradation in speech quality [9]–[12]. To overcome these problems with parallel data, this paper focus on a non-parallel VC method, which does *not* require any parallel utterances, transcriptions, or time alignment.

Developing non-parallel VC framework is challenging task because of the problems associated with the training conditions using non-parallel data in deep learning architectures. However, attempts have been made to develop many non-parallel VC frameworks in the past decade. For example, Maximum Likelihood (ML)-based approach proposed by Ye *et al.* [13], speaker adaptation technique by Mouchtaris *et al.* [14], GMM-based VC method using Maximum a posteriori (MAP) adaptation technique by Lee *et al.* [15], iterative alignment method by Erro *et al.* [16], Automatic Speech Recognition (ASR)-based method [17], speaker verification based method using i-vectors [18], and many other frameworks [6], [19]–[27]. Recently, a method using Conditional Variational Autoencoders (VAEs) [28] was proposed for non-parallel VC in [22], [24]. One powerful framework that can potentially overcome the weakness of VAEs involves GANs [29]. While GAN-based methods originally applied for image-translation problems, these methods have been also employed with noteworthy success for various speech technology related applications [30]–[33]. In GANs-based methods, Cycle-consistent Adversarial Network (CycleGAN)-VC is one of state-of-the-art methods in non-parallel VC task proposed by Kaneko *et al.* in [23]. In addition, StarGAN is proposed for many-to-many VC, which uses target speaker indication via one-hot vector [34]. However, we aim to improve one-to-one VC by using the novel concept of speaker *style* transfer.

This paper addresses the problem of non-parallel VC using adversarial training. In CycleGAN, we need feature-based mapping technique for both style-transfer and content preservation. Moreover, CycleGAN consists of two different generators and discriminators for non-parallel VC task. To overcome this complexity in such cross-domain architectures (i.e., CycleGAN, Discover GAN, DualGAN, etc.), we introduce a novel Adaptive Generative Adversarial Network (AdaGAN) with less complexity in terms of computation and time. AdaGAN has only one generator which primarily does

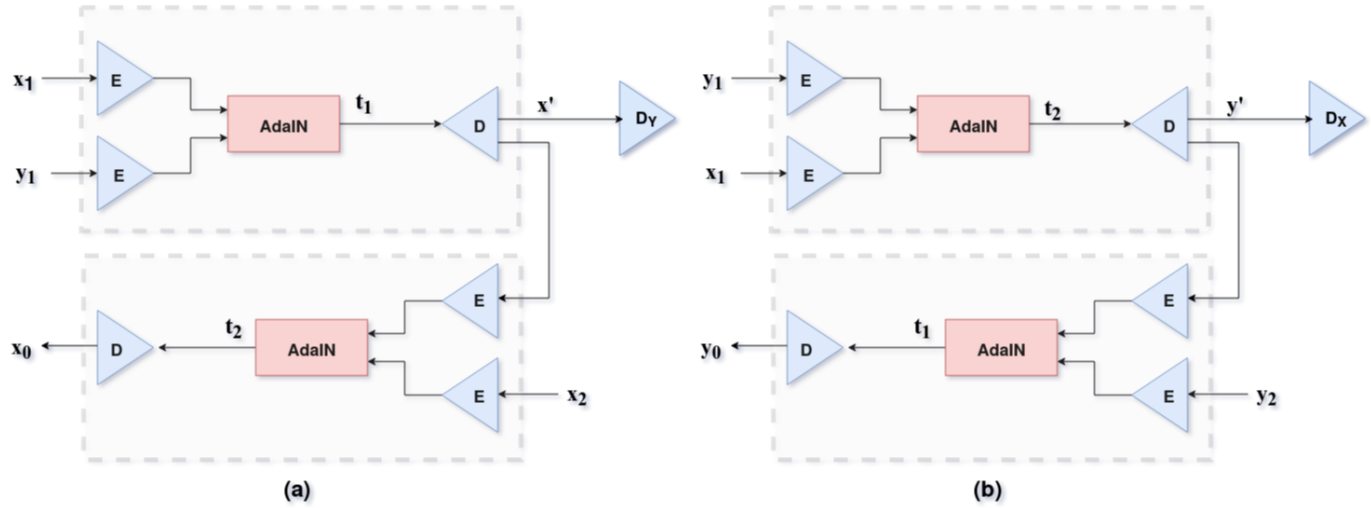


Fig. 1: Proposed AdaGAN architecture with transferring the style: (a) y to x and (b) x to y.

style transfer using Adaptive Instance Normalization (AdaIN) and two discriminators which are responsible for adversarial training of generator. The key idea behind AdaGAN is to encapsulate the style of a target speaker into a source speaker's speech which retains the same content without doing feature-based mapping for linguistic information.

Experimental results show that our proposed AdaGAN outperforms CyleGAN in terms of subjective as well as objective results on VC challenge 2018 corpus for non-parallel VC. We observe that the AdaGAN yields objective results that are comparable to the CycleGAN, and are superior in terms of subjective evaluation. We also performed statistical analysis in terms of t-SNE visualisation, comparisons of spectrograms, and Teager Energy Operator (TEO) profile-based analysis to show the superiority of AdaGAN over CycleGAN.

II. Proposed Method

In this section, we introduce the *AdaIN* for style transfer, and give description about how we encapsulate AdaIN in AdaGAN. Furthermore, we compare the AdaGAN with the state-of-the-art CycleGAN architecture for parallel-data free one-to-one VC.

A. Adaptive Instance Normalization (AdaIN)

The concept of *AdaIN* was first introduced in computer vision for doing arbitrary style transfer on the image-to-image translation task [35]. In this paper, we first time introduce *AdaIN* for VC task. *AdaIN* takes features x and y as content and style input, respectively. It will align the mean and variance of the feature x in such a way that it will match with mean and variance of feature y. However, *AdaIN* will adjust the parameters by considering the input style feature y since it does not have any learnable parameters. The mathematical form of *AdaIN* is defined as:

$$AdaIN(x, y) = \sigma(y)\Big(\frac{x - \mu(x)}{\sigma(x)}\Big) + \mu(y). \qquad (1)$$

From eq. 1, we can infer that *AdaIN* first normalizes input content features x, and scales back based on mean and variances of input style features y. Intuitively, let us consider features of a particular speaker contains specific style. As a result, the output produced by *AdaIN* have the high average activation for the features which are responsible for style while preserving linguistic content.

B. AdaGAN

AdaGAN uses non-parallel corpus to learn the mapping function between source speaker (i.e., $x \in X$) to a target speaker (i.e., $y \in Y$), where X and Y are the data distributions for features of a source, and target speakers' speech, respectively. AdaGAN uses one generator (i.e., G) and two discriminators, D_X and D_Y to identify styles of speaker x and y, respectively. Furthermore, the generator is divided into three sub-parts, namely, encoder (E), *AdaIN*, and decoder (D). Here, the same E will extract the features F_X and F_Y from the inputs x and y, respectively. Both of these features will be treated as content and style features as input to the *AdaIN*, respectively. The output of the *AdaIN* will be scaled features, F_X according to the features, F_Y. Let this be denoted as t (t_1 and t_2 in Fig. 1). In Fig. 1, t_1 denotes scaling of feature F_X according to feature F_Y, and t_2 denotes scaling of feature F_Y according to feature F_X. Now, this t will be given as input to the D, and this D will generate output features, x'. This converted feature x' should be similar on style with the domain Y, and it should preserve the content of the input features, x. This converted features x' are provided to discriminator D_Y, which ensure adversarial training. Overall, our goal is to make E to extract the features in such a way that the content will be independent of the distribution, however, style of the speaker will be preserved in the distribution. Hence, by scaling the input based on target in *AdaIN*, the style distribution of the speaker should be transferred, and content should be conserved. Moreover, D will learn to generate back from encoded input feature, and D_X plays a role for adversarial

training for transferring style of x to y. Fig. 1 shows the proposed AdaGAN architecture for style transfer.

To achieve the goal, we train AdaGAN as described in Fig. 1. Using encoder, we extracted the features F_{X_1}, F_{X_2}, F_{Y_1}, and F_{Y_2} of x_1, x_2, y_1, and y_2, respectively. After this, we have used $AdaIN$ to transfer the style of y_1 to x_1, and the output from $AdaIN$ is given to decoder to decode this scaled feature to $x^{'}$. Now, $x^{'}$ will be passed from the same encoder again, and this encoded features alongside the F_{X_2} will be given as the input to $AdaIN$. Instead of taking same input x_1, we have used x_2 to transfer the speaking style of speaker X since E will become bias to the inputs by using x_1 again, and our proposed cycle-consistent loss will not be effective. Moreover, the same decoder will decode back from this newly scaled feature, and this should be similar to the input x_1. To transfer the style of x_1 into y_1, we do the same as described above (see Fig. 1 (b)). Since our task is to encapsulate the target speaker's style into source speaker's speech while preserving content, we explore different loss functions to achieve this. In addition, all of our loss functions (except adversarial loss) incorporates L_1 norm instead of L_2 norm as suggested in [36].

Adversarial loss: This loss is used to make the generated or converted speech indistinguishable from the original target voice. This loss function can be mathematically formulated as:

$$\begin{aligned}\mathcal{L}_{G_{XY}} &= \mathbb{E}_{x'\sim X}[(\log(D_X(y')-1)^2] \\ &+ \mathbb{E}_{y'\sim Y}[\log(D_Y(x')-1)^2],\end{aligned} \tag{2}$$

$$\begin{aligned}\mathcal{L}_{D_{XY}} &= \mathbb{E}_{x_1\sim X}[(\log(D_X(x_1)-1)^2] \\ &+ \mathbb{E}_{y_1\sim Y}[\log(D_Y(y_1)-1)^2].\end{aligned} \tag{3}$$

Cycle Consistency Loss: By using only adversarial loss, we may loose linguistic information from the converted voice. To make sure that the generator G learns to keep the linguistic information, we use L_1 norm as cycle consistency loss, and can be described as:

$$\begin{aligned}\mathcal{L}_{cyc} &= \mathbb{E}_{\{x_1,x_2\}\sim X}[\|G(G(x_1,y_1),x_2)-x_1\|_1] \\ &+ \mathbb{E}_{\{y_1,y_2\}\sim Y}[\|G(G(y_1,x_1),y_2)-y_1\|_1].\end{aligned} \tag{4}$$

In Fig. 1, $G(x,y)$ is denoted by x', and $G(x',x)$ is denoted by x_0, and it should be the same as x. Similarly, $G(y,x)$ is denoted by y' and $G(y',y)$ is denoted by y_0, and it should be the same as y.

Style Transfer Loss: To preserve the transferred style of the input speech during *AdaIN*, we use following L_1 norm, i.e.,

$$\mathcal{L}_{C_{X\to Y}} = \|E(D(t_1)) - t_1\|_1, \tag{5}$$

where t_1 is the output of *AdaIN* for given inputs x_1, and y_1. By using this loss, we are giving the direction to the encoder in which it should preserve the style between t_1 and again encoded feature of x' by keeping the encoded features' distribution similar to each other. This is the only style which is similar between t_1 and $F_{X'}$, whereas content will be different between them. This raise the question regarding the similarity of content. Therefore, we use the following loss function to solve this issue.

Content Preserve Loss: To preserve the content of the input speech during *AdaIN*, we use following L_1 norm, i.e.,

$$\mathcal{L}_{S_{X\to Y}} = \|AdaIN(E(D(t_1))) - F_{X_1}\|_1, \tag{6}$$

where F_{X_1} is the encoded feature of input x_1 and $AdaIN(E(D(t_1)))$ is nothing but t_2 (shown in fig 1).

For given target encoded feature F_{Y_1}, we use *AdaIN* to get t_1. After this, we simply use decoder D to generate x'. In addition, we use this generated x' and x_2 as input to the *AdaIN*. Output of this *AdaIN*, namely t_2, should be the similar to the encoded feature of x_1 which is F_{X_1} in terms of content and style both because both features contains the encoded versions of the different input features with same content and from the same speaker, one or in another way. Hence, this loss ensures that content should be preserved during the whole process.

Final Objective Loss Function: The overall objective function of AdaGAN can be mathematically defined as:

$$\begin{aligned}\mathcal{L}_{total} =& \mathcal{L}_{G_{XY}} + \mathcal{L}_{D_{XY}} + \lambda_1\mathcal{L}_{cyc} + \lambda_2\mathcal{L}_{C_{X\to Y}} + \\ & \lambda_3\mathcal{L}_{C_{Y\to X}} + \lambda_4\mathcal{L}_{S_{X\to Y}} + \lambda_5\mathcal{L}_{S_{Y\to X}},\end{aligned} \tag{7}$$

where λ_1, λ_2, λ_3, λ_4, and λ_5 are the hyperparameters. These parameters controls the relative importance of each loss w.r.t. each other. We have used $\lambda_1 = 10$, $\lambda_2 = 2$, $\lambda_3 = 2$, $\lambda_4 = 3$, and $\lambda_5 = 3$ during the experiments.

C. CycleGAN vs. AdaGAN

In CycleGAN for VC, we use two different generators to find the mapping function between the features of the source speaker's speech and the features of the target speaker's speech. Whereas, AdaGAN uses a single generator to do the same task. In CycleGAN, generators are trained for only convert first speaker voice to the second speaker not vice-versa. Hence, generators are doing only a specific task in CycleGAN. In contrast, AdaGAN uses one generator to convert first to second speaker's and second to first speaker's voice. The reason is, an encoder in AdaGAN has learned in such a way that it can extract the content and style of both the speakers. AdaIN can transfer the style very easily, and decoder is trained to decode any content with any style. There is no concept of extracting speaker style and content in CycleGAN since it uses a generator to convert voice directly without considering content or style. When we calculate the number of different parameters (i.e., weights and biases) that are needed to learn in CycleGAN and AdaGAN for each iterations are 2,227,282 and 2,185,770 respectively. Hence, we can clearly see that AdaGAN requires 41,512 less parameters to learn w.r.t. to CycleGAN. In addition, AdaGAN takes on an average around 860 seconds training time for single source-target pair. However, CycleGAN takes 1220 seconds for training in same conditions. The training was done on PC containing 16GB RAM and GTX1070Ti graphic card. Hence, with less complexity, it still gives comparable results in objective assessment and outperforms in subjective evaluation (as shown in section IV).

III. Experimental Setup

A. Database

The experiments are evaluated on the Voice Conversion Challenge (VCC) 2018 database [37]. The statistics of the database are given in [37]. The database is designed to provide two cases, namely, Hub task (i.e., parallel training data), and Spoke task (i.e., non-parallel training data) [37]. VC systems were developed among 16 speaker-pairs for each the Hub and the Spoke task. Statistics of the databases are shown in Table I.

TABLE I: Statistics of the VCC 2018 database. After [37].

Task	Speaker	No. of Speakers		No. of Utterances	
		Male	Female	Training	Testing
Spoke	Source	2	2	81	35
	Target	2	2	81	35

The *40*-dimensional (dim) Mel Cepstral Coefficients (MCCs) (including the 0^{th} coefficient) are extracted from the speeches of source and the target speakers with 25 ms window and 5 ms frameshift. For analysis-synthesis, we have used AHOCODER [38].

B. Architecture Details

In this paper, generators in CycleGAN follow the identical architecture with the three hidden layers. Each hidden layer contains 512 neurons with Rectified Linear Unit (ReLU) activation, whereas the output layer has a linear activation function. Generator of AdaGAN consists of the encoder and decoder as suggested in Fig. 1. Both the encoder and decoder follow the identical architecture with the two hidden layers, and each hidden layer contains 512 neurons with ReLU activation function. The discriminators of the CycleGAN and AdaGAN also have three hidden layers, with ReLU activation function, whereas the output layer has sigmoid activation function. CycleGAN and AdaGAN are trained for 100 epochs, using an effective batch size of 1000 frames as suggested in [30]. The parameters are optimized using Adam optimization, with a learning rate of 0.0001.

IV. Experimental Results

Objective, statistical and subjective evaluations are carried out on VCC 2018 database. In the objective evaluation, we analyze our proposed model in terms of Mel Cepstral Distortion (MCD) score and flexibility of architecture. In statistical analysis, we have analyzed t-SNE visualization and spectrogram comparison. In the subjective evaluation, we analyze our model via subjective test given by the humans in terms of speaker similarity, sound quality, and naturalness. Samples are presented here [1].

[1]https://sites.google.com/view/adagan

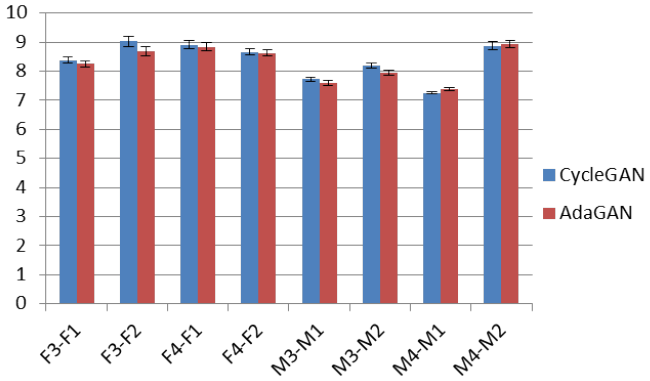


Fig. 2: MCD analysis of the different systems based intra-gender non-parallel VC task along with 95% confidence interval.

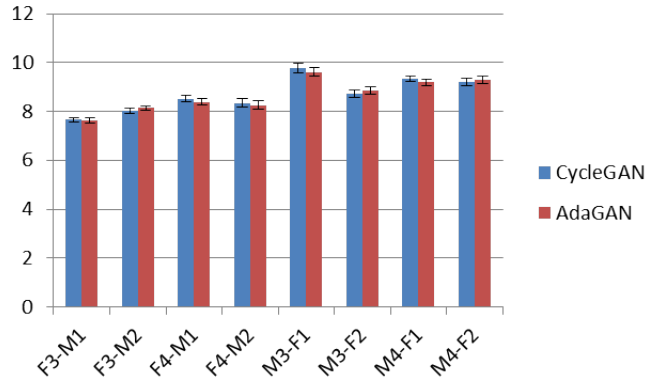


Fig. 3: MCD analysis of the different systems based inter-gender non-parallel VC task along with 95% confidence interval.

A. Objective Evaluation

1) Mel Cepstral Distortion (MCD): We have applied MCD-based objective measures to analyze the effectiveness of the non-parallel VC systems. The traditional MCD measure is used here which is given by [2]:

$$MCD \text{ [in dB]} = \frac{10}{ln10}\sqrt{2\sum_{i=1}^{40}(m_i^t - m_i^c)^2}\ , \qquad (8)$$

where m_i^t and m_i^c are the i^{th} MCCs of the reference, and converted voice. In particular, m_i^t and m_i^c are the i^{th} MCCs of the reference speech of target speaker and the converted voice in the case of VC system. Since MCD is the distance between the converted and the reference cepstral features, a system that is having lesser MCD is considered as a better system. We analyze both the architectures, AdaGAN and CycleGAN in terms of MCD score for intra-gender and inter-gender VC. We compare the MCD score for intra-gender and inter-gender VC, as shown in Fig. 2 and Fig. 3, respectively. We can observe that AdaGAN got comparable MCD score w.r.t. CycleGAN for almost all the 16 systems. In many systems, AdaGAN gives almost 3.7 % (on an average) reduction in MCD compared to the CycleGAN. For example, M3-M2, F3-F2, M3-F1, to name a few. Here, less MCD score indicates

that the content preservation is better in AdaGAN, and AdaIN helps to encapsulate the style of a target speaker in a better way. Overall, we can say that speech intelligibility is more in AdaGAN, and we can also see this via subjective evaluation.

2) Flexibility: We define the flexibility of AdaGAN in terms of the ability of encapsulating the transferred speakers' style. Our proposed method allows us to control the style and content without changing the training procedure. To do this, we use the following function before giving the output of $AdaIN$ to the decoder:

$$T(F_X, F_Y, \alpha) = \alpha * F_X + (1 - \alpha) * AdaIN(F_X, F_Y), \quad (9)$$

where hyperparameter α controls the trade-off between style and content of the speech, and F_X and F_Y are encoded features of x and y, respectively. To evaluate the flexibility of AdaGAN, we plot the graph of Global Variance (GV) *vs.* index of Mel Cepstral Coefficient. From Fig. 4, we can see that for $\alpha = 0.3$, GV (i.e., green colour line) is slightly more close to the GV of the target (i.e., blue colour line), compared to other values of α. Here, we define the value of α empirically. Furthermore, we found the mean distance for each α value based on GV curve, we got the 0.51, 0.55 and 0.54 for α values 0.3, 0.1 and 0.6, respectively. This result also supports our argument that AdaGAN trained with $\alpha = 0.3$ results in more accurate model.

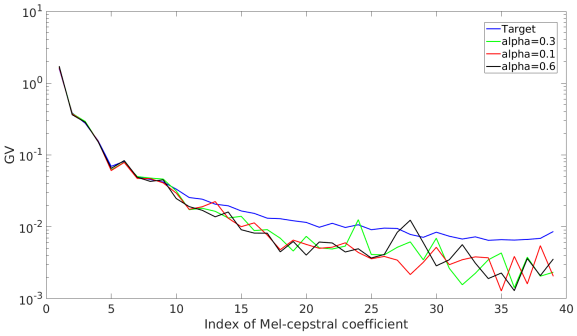
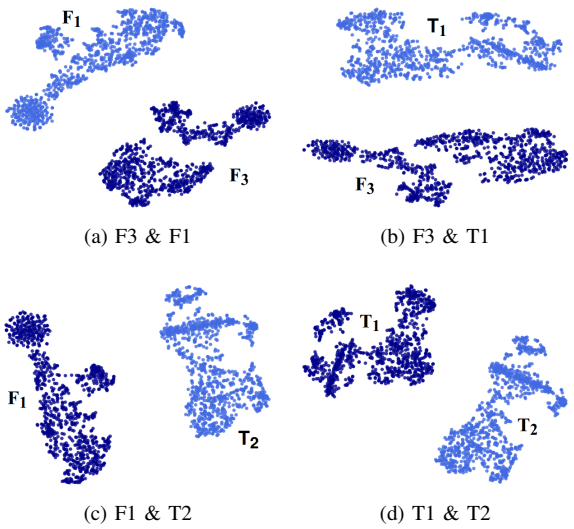


Fig. 4: Global Variance(GV) plots with respect to different α value.

B. Statistical Analysis

We have done statistical analysis which proves that AdaGAN is able to do speaking style transfer efficiently compare to CycleGAN for non-parallel one-to-one VC tasks. We prove this claim through t-SNE visualization, spectrographic analysis, and TEO profile based visualization.

1) t-SNE Visualization: Our main objective in VC is to transfer the speaking style of one speaker to another speaker. To do this, we have used encoder to encode the input MCC features in such a way that style of speaker will be captured via distribution, and linguistic content will be independent of this distribution (i.e., normalized encoded features will represent the content). This is achieved by the loss functions which use AdaIN properties to do the same. However, it is difficult to train neural network-based model to learn style of speaking. For example, inappropriate choice of loss functions in encoder degrades the performance of whole system. Hence, to ensure



(a) F3 & F1 (b) F3 & T1

(c) F1 & T2 (d) T1 & T2

Fig. 5: t-SNE visualizations of different feature pairs.

that our architecture is trained properly, we visualize the output of encoded features of different speakers' voices, which are taken before applying AdaIN, and after applying the AdaIN. To visualize the outputs of 512 dimensional encoder, we have used t-SNE to reduce the feature dimensions to two [39]. For these set of experiments, we have used VCC2SF3 and VCC2SF1 speakers data on which AdaGAN is trained.

From Fig. 5(a), we can observe that encoded features F_3 and F_1 are well separated, and clustered together w.r.t. the speaking styles of the speakers VCC2SF3 and VCC2SF1. If we transfer the distribution of F_1 to F_3 via AdaIN which is denoted as t_1 in Fig. 1(a), t_1 and F_3 must be well separated which is observed in Fig. 5(b). This can also true for F_1 and t_2 as shown in Fig. 5(c). In addition, t_1 and t_2 both should be separated just like F_1 and F_3, which is shown in Fig. 5(d).

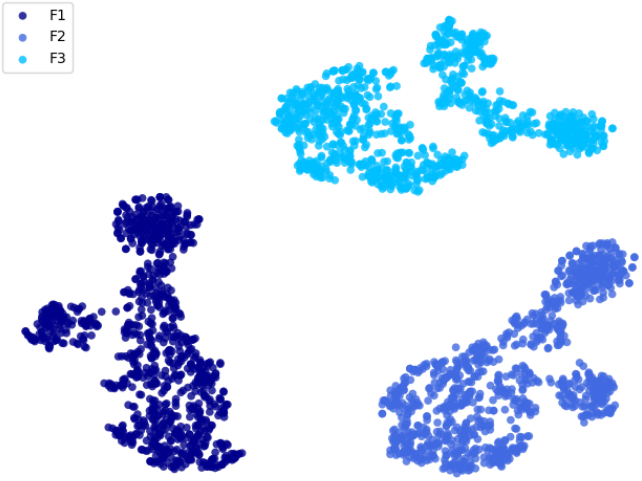
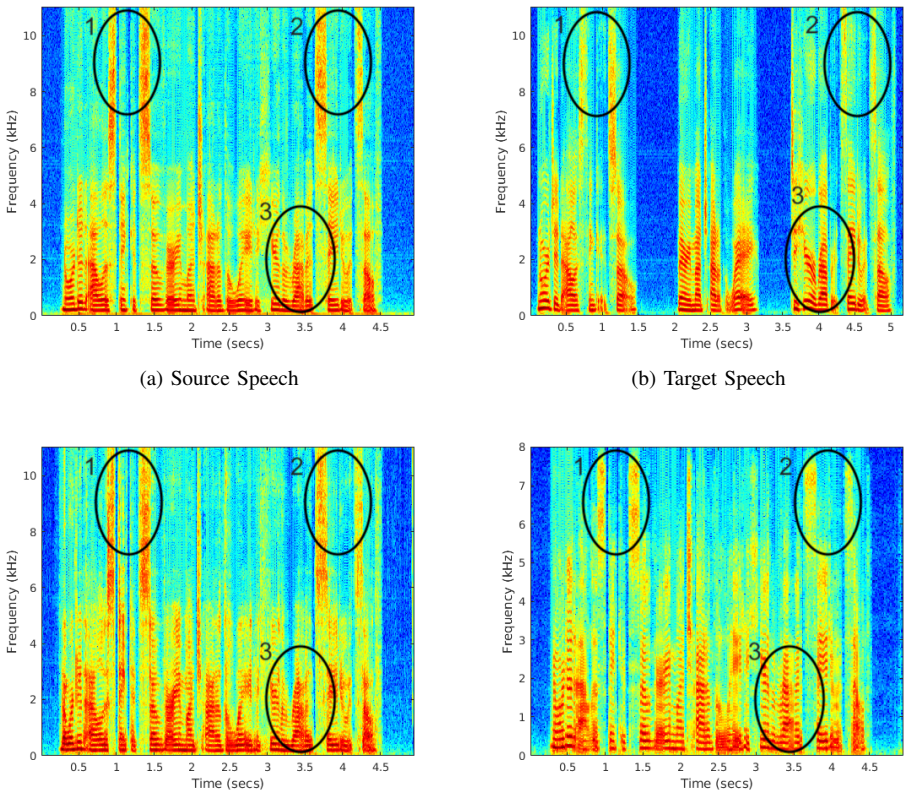


Fig. 6: t-SNE visualization of encoded features of three speakers VCC2SF1, VCC2SF2, and VCC2SF3.

As shown in Fig. 6, AdaGAN leads us to the possibility of many-to-many VC. For example, AdaGAN is trained



(a) Source Speech (b) Target Speech

(c) Converted Speech using CycleGAN (d) Converted Speech using AdaGAN

Fig. 7: Spectrograms for (a) source speech, (b) target speech, (c) Converted speech using CycleGAN, and (d) Converted speech using AdaGAN for the utterance "Nor did Alas think it so, very much out of the way, to hear the rabbit to say itself".

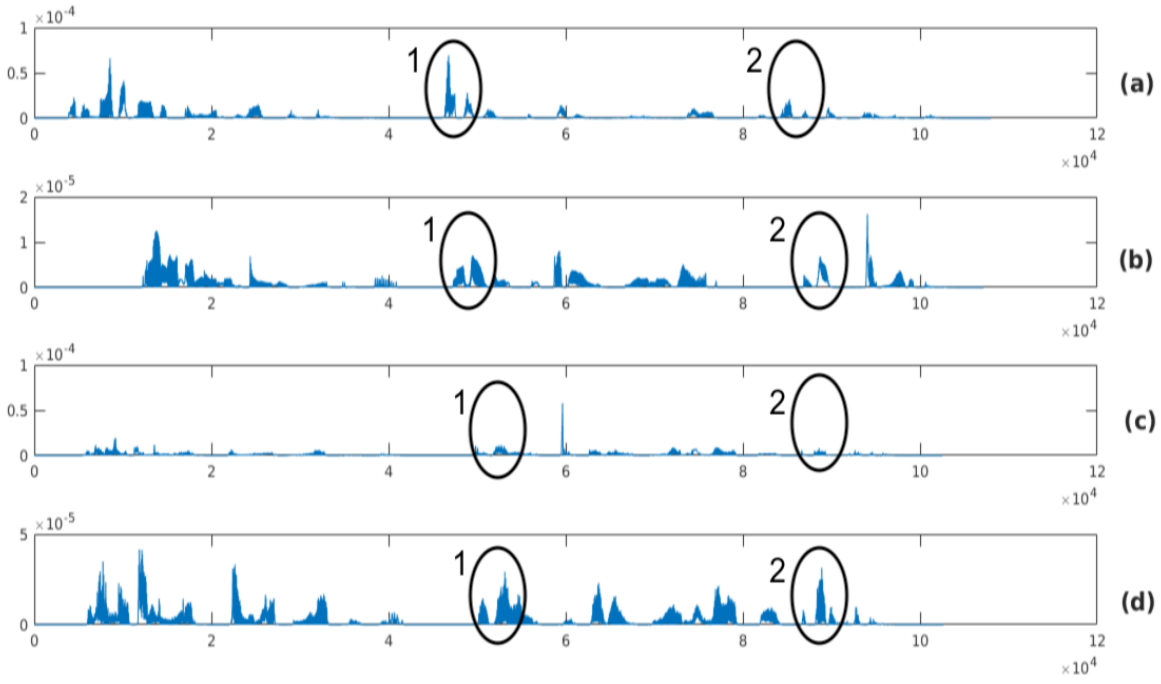


Fig. 8: 1^{st} subband TEO profile visualization for VCC2SM3 to VCC2TF1 inter-gender VC task. Here, (a), (b), (c), and (d) are the TEO profiles of source speech, target speech, generated voice using CycleGAN, and generated voice using AdaGANGAN, respectively.

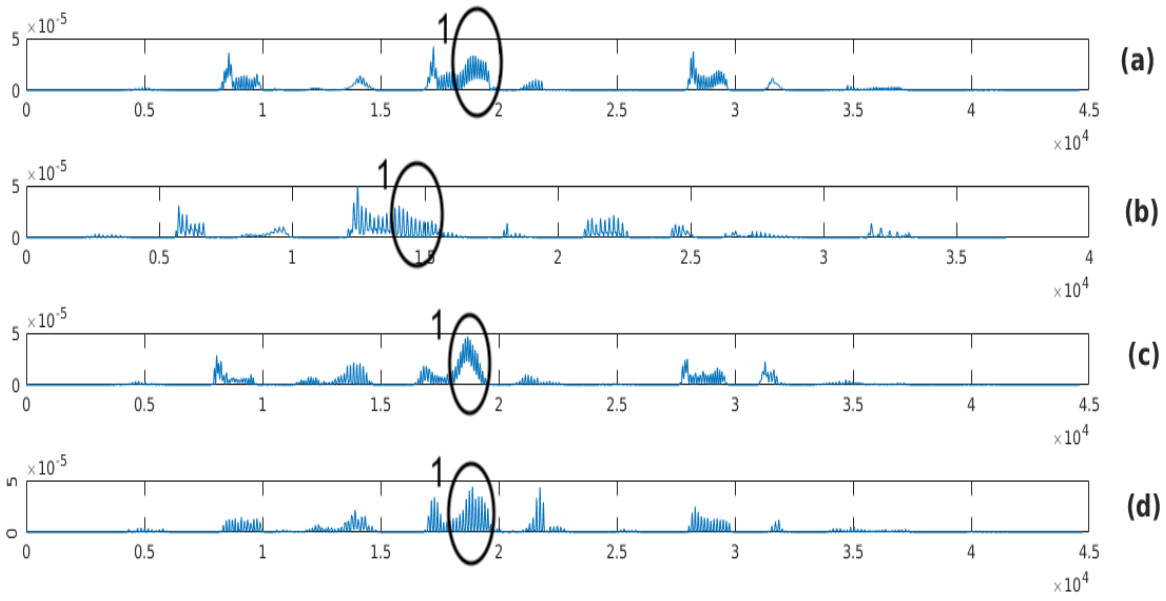


Fig. 9: 3^{rd} subband TEO profile visualization for VCC2SM3 to VCC2TM1 intra-gender VC task. Here, (a), (b), (c), and (d) are the TEO profiles of source speech, target speech, generated voice using CycleGAN, and generated voice using AdaGAN, respectively.

for VCC2SF3-VCC2TF1 conversion task. Now, we extract the features from encoder for another unseen speaker (i.e., VCC2SF2), and do the t-SNE visualization for these three different speakers. We observed that they are clustered w.r.t. the each speakers' speaking style (as shown in Fig. 6). This analysis bolsters the idea of many-to-many VC using AdaGAN.

2) Spectrographic Analysis: Form Fig. 7, we can observe that AdaGAN captures the speaking style more efficiently compare to CycleGAN at higher frequency regions. These effects are shown by ellipses 1 and 2 in Fig. 7. About the lower frequency regions in target speech, the spectral energy densities are much lower compared to the source speech. This is shown in Fig. 7 by 3^{rd} ellipse region. From this, we can see that CycleGAN is preserving spectral energy density, whereas, AdaGAN modifies it like the target speech. Here, we have shown the spectrographic analysis for only one speech utterance.

We can conclude from spectrographic analysis that AdaGAN is more efficient in parallel-data free VC compare to CycleGAN. The reason behind this is related to cycle consistent loss and identity loss in the training of CycleGAN for preserving linguistic information. Hence, the generator of CycleGAN is learning to modify input speech, to make it looks like target speech via changing in small amount of spectral energy density. On contrary, AdaGAN is able to modify spectral energy density of input speech more efficiently due to the different loss functions alongside the cycle consistent loss. Hence, AdaGAN is more efficient in parallel-data free VC compare to CycleGAN.

3) Teager Energy Operator (TEO) Profile-based Analysis: We have applied Gabor filterbank to source, target, CycleGAN, and AdaGAN generated speeches. We have used Gabor filterbank with linearly spaced central frequencies [40]–[43] to get the number of subbands outputs (i.e., total 40 subbands). The Gabor filterbank have optimal joint time-frequency resolution [44], [45]. Now each of these subband signals are used to compute TEO profile. After extracting TEO profile of each of these subbands, we compare them visually. We have ignored higher subbands' TEO profiles because they does not capture much information. Hence, we have considered first 10 subband-based TEO profiles. In Fig. 8 and Fig. 9, we have shown two different subband TEO profiles for two separate conversion tasks (i.e., intra-gender and inter-gender VC). In Fig. 8, we can clearly observe that TEO bumps are very less for source speech, whereas TEO bumps are more in the case of target speech. However, TEO bumps in CycleGAN generated speech is less in numbers, while AdaGAN increases these bumps to make speech more similar to target speech. These effects can be seen from 1^{st} and 2^{nd} ellipses in Fig. 8. In addition, the similar kind of case was observed for intra-gender VC (shown in Fig. 9). This analysis strengthens our conclusion that AdaGAN is more efficient in parallel-data free VC compare to CycleGAN. Moreover, AdaGAN can do style transfer in each of the frequency subbands.

C. Subjective Evaluation

Comparative subjective analysis test, namely, ABX, AB, and Mean Opinion Score (MOS) has been conducted for subjective evaluations. Total of 30 subjects (7 females and 23 males) between 18-30 years of age and with no known hearing im-

pairments took part in the subjective test. We have used high-quality Sennheiser headphones for subjective evaluation. Here, we randomly played the same utterances from two different systems (i.e., AdaGAN and CycleGAN), and asked subjects to decide which one is better in terms of sound quality. We also played different generated voices by CycleGAN and AdaGAN, and asked subjects to rate them in terms of naturalness between 1-5 (1 being the lowest and 5 being the highest). In addition, we randomly played three speeches (target, CycleGAN and AdaGAN generated) containing same utterances, and asked subjects to choose which one is more similar to the target speaker or both are same.

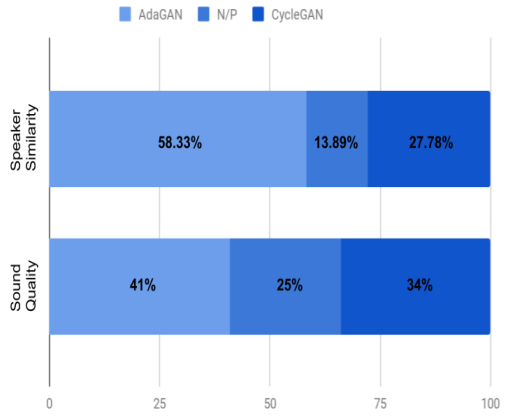


Fig. 10: Speaker similarity (ABX) and Sound quality (AB) tests analysis for the CycleGAN and AdaGAN.

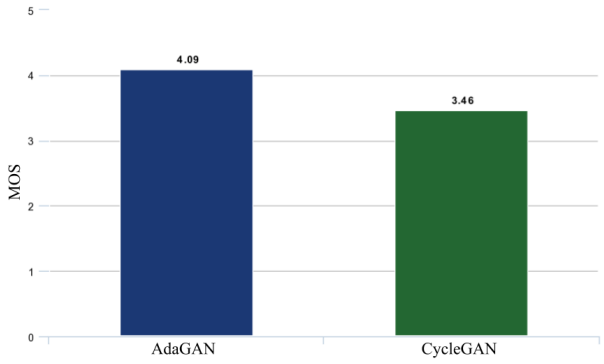


Fig. 11: MOS score analysis for naturalness for AdaGAN and CycleGAN, respectively.

Results of the Sound quality (AB) test, Naturalness (MOS) test, and Speaker Similarity (ABX) test obtained from the total 240 samples for each test. Results of AB tests are shown in Fig. 10. We observed that the proposed AdaGAN is 41% (on an average) times more preferred over the CycleGAN by the subjects in terms of sound quality. While in case of speaker similarity, AdaGAN preferred 58.33% times more over CycleGAN. In addition, results of MOS tests are shown in Fig. 11. Here, we can also observed that the proposed AdaGAN has MOS score for naturalness of 4.09 while CycleGAN has only 3.46. Therefore, we conclude that AdaGAN outperforms CycleGAN in various subjective tests.

V. Summary and Conclusions

In this paper, we propose novel AdaGAN for parallel-data free one-to-one VC. Moreover, we analyzed our proposed architecture w.r.t. current state-of-the-art CycleGAN method. We know that the main aim of VC is to convert source speaker's voice into target speaker's voice while preserving linguistic content. To achieve this, CycleGAN uses two generators and two discriminators for feature-based mapping between two domains. AdaGAN does the same task with less computational complexity and training time compare to CycleGAN. AdaGAN transfer the style of the target speaker into the voice of a source speaker without using any feature-based mapping between linguistic content of source speaker's speech. For this task, AdaGAN uses only one generator, and two discriminators which leads to less complexity. We have done objective and subjective analysis on VCC 2018 corpus to show the efficiency of proposed method. We can clearly see that AdaGAN gives comparable results in terms of objective evaluation, whereas it yields superior results in subjective analysis compare to CycleGAN. In addition, we have done statistical analysis in order to bolster our objective and subjective evaluations. Our future work will be directed towards modifying the AdaGAN for many-to-many non-parallel VC, and planning to explore high-quality vocoders, namely, World, WaveNet for further improvement in voice quality. The perceptual difference observed between the estimated and the ground truth indicates the need of exploring better objective function that can perceptually optimize the network parameters of GAN-based architectures, which also forms our future work.

VI. Acknowledgements

The authors would like to thank the authorities of DA-IICT, Gandhinagar, India and Ministry of Electronics and Information Technology (MeitY), New Delhi, Govt. of India for their kind support to carry out this research work. Special thanks to all the subjects, who took part in the subjective evaluation.

References

[1] Seyed Hamidreza Mohammadi and Alexander Kain, "An overview of voice conversion systems," *Speech Communication*, vol. 8, pp. 65–82, 2017.

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

[3] Yannis Stylianou, Olivier Cappé, and Eric Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Transactions on Speech and Audio Processing*, vol. 6, no. 2, pp. 131–142, 1998.

[4] Seyed Hamidreza Mohammadi and Alexander Kain, "Voice conversion using deep neural networks with speaker-independent pre-training," in *2014 IEEE Spoken Language Technology Workshop (SLT)*, USA, 2014, IEEE, pp. 19–23.

[5] Yuki Saito, Shinnosuke Takamichi, and Hiroshi Saruwatari, "Voice conversion using input-to-output highway networks," *IEICE Transactions on Information and Systems*, vol. 100, no. 8, pp. 1925–1928, 2017.

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

[7] Takuhiro Kaneko, Hirokazu Kameoka, Kaoru Hiramatsu, and Kunio Kashino, "Sequence-to-sequence voice conversion with similarity metric learned using generative adversarial networks," in *INTERSPEECH*, Sweden, 2017, pp. 1283–1287.

[8] Zhizheng Wu, Tuomas Virtanen, Eng Siong Chng, and Haizhou Li, "Exemplar-based sparse representation with residual compensation for voice conversion," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 10, pp. 1506–1521, 2014.

[9] E. Helander, J. Schwarz, J. Nurminen, H. Silen, and M. Gabbouj, "On the impact of alignment on voice conversion performance," in *INTERSPEECH*, Brisbane, Australia, 2008, pp. 1–5.

[10] Nirmesh J. Shah and Hemant A. Patil, "A novel approach to remove outliers for parallel voice conversion," *Computer Speech & Language*, 2019.

[11] Sushant V Rao, Nirmesh J Shah, and Hemant A Patil, "Novel pre-processing using outlier removal in voice conversion," in *Speech Synthesis Workshop (SSW)*, Sunnyvale, CA, USA, 2016, pp. 134–139.

[12] Nirmesh J. Shah and Hemant A. Patil, "Analysis of features and metrics for alignment in text-dependent voice conversion," in *International Conference on Pattern Recognition and Machine Intelligence (PReMI), ISI, Kolkata: B. Uma Shankar et. al., Lecture Notes in Computer Science (LNCS), Springer*, 2017, vol. 10597, pp. 299–307.

[13] Hui Ye and Steve Young, "Quality-enhanced voice morphing using maximum likelihood transformations," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1301–1312, 2006.

[14] Athanasios Mouchtaris, Jan Van der Spiegel, and Paul Mueller, "Non-parallel training for voice conversion based on a parameter adaptation approach," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 3, pp. 952–963, 2006.

[15] Chung-Han Lee and Chung-Hsien Wu, "MAP-based adaptation for speech conversion using adaptation data selection and non-parallel training," in 9^{th} *International Conference on Spoken Language Processing*, USA, 2006.

[16] Daniel Erro, Asunción Moreno, and Antonio Bonafonte, "INCA algorithm for training voice conversion systems from nonparallel corpora," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 5, pp. 944–953, 2010.

[17] Feng-Long Xie, Frank K Soong, and Haifeng Li, "A KL divergence and DNN-based approach to voice conversion without parallel training sentences.," in *INSTERSPEECH*, USA, 2016, pp. 287–291.

[18] Tomi Kinnunen, Lauri Juvela, Paavo Alku, and Junichi Yamagishi, "Non-parallel voice conversion using i-vector PLDA: Towards unifying speaker verification and transformation," in *2017 IEEE ICASSP*, USA, 2017, IEEE, pp. 5535–5539.

[19] Ling-Hui Chen, Zhen-Hua Ling, Li-Juan Liu, and Li-Rong Dai, "Voice conversion using deep neural networks with layer-wise generative training," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 22, no. 12, pp. 1859–1872, 2014.

[20] Toru Nakashika, Tetsuya Takiguchi, and Yasuo Ariki, "Voice conversion based on speaker-dependent restricted boltzmann machines," *IEICE TRANSACTIONS on Information and Systems*, vol. 97, no. 6, pp. 1403–1410, 2014.

[21] Merlijn Blaauw and Jordi Bonada, "Modeling and transforming speech using variational autoencoders," in *INTERSPEECH*, USA, 2016, pp. 1770–1774.

[22] Chin-Cheng Hsu, Hsin-Te Hwang, Yi-Chiao Wu, Yu Tsao, and Hsin-Min 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)*, Korea, 2016, IEEE, pp. 1–6.

[23] Takuhiro Kaneko and Hirokazu Kameoka, "Parallel-data-free voice conversion using cycle-consistent adversarial networks," *arXiv preprint arXiv:1711.11293*, 2017.

[24] Yuki Saito, Yusuke Ijima, Kyosuke Nishida, and Shinnosuke Takamichi, "Non-parallel voice conversion using variational autoencoders conditioned by phonetic posteriorgrams and d-vectors," in *2018 IEEE ICASSP*, Canada, 2018, IEEE, pp. 5274–5278.

[25] Nirmesh J. Shah, Maulik Madhavi, and Hemant A. Patil, "Unsupervised vocal tract length warped posterior features for non-parallel voice conversion," in *INTERSPEECH*, Hyderabad, India, 2018, pp. 1968–1972.

[26] Nirmesh J. Shah, Sreeraj R., Neil Shah, and Hemant A. Patil, "Novel inter mixture weighted GMM posteriorgram for DNN and GAN-based voice conversion," in *APSIPA*, Hawaii, USA, 2018, IEEE, pp. 1776–1781.

[27] Nirmesh J. Shah and Hemant A. Patil, "Effectiveness of dynamic features in inca and temporal context-inca," in *INTERSPEECH*, Hyderabad, India, 2018, pp. 711–715.

[28] Diederik P Kingma and Max Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[29] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems (NIPS)*, Canada, 2014, pp. 2672–2680.

[30] Neil Shah, Nirmesh J. Shah, and Hemant A. Patil, "Effectiveness of generative adversarial network for non-audible murmur-to-whisper speech conversion," in *INTERSPEECH*, Hyderabad, India, 2018, pp. 3157–3161.

[31] Nirmesh Shah, Mihir Parmar, Neil Shah, and Hemant A. Patil, "Novel MMSE DiscoGAN for cross-domain whisper-to-speech conversion," in *Machine Learning in Speech and Language Processing (MLSLP) Workshop*, Google Office, Hyderabad, India, 2018, pp. 1–3.

[32] Daniel Michelsanti and Zheng-Hua Tan, "Conditional generative adversarial networks for speech enhancement and noise-robust speaker verification," in *INTERSPEECH*, Stockholm, Sweden, 2017, pp. 3642–3646.

[33] Yuki Saito, Shinnosuke Takamichi, and Hiroshi Saruwatari, "Statistical parametric speech synthesis incorporating generative adversarial networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 1, pp. 84–96, 2018.

[34] Hirokazu Kameoka, Takuhiro Kaneko, Kou Tanaka, and Nobukatsu Hojo, "Stargan-vc: Non-parallel many-to-many voice conversion with star generative adversarial networks," *arXiv preprint arXiv:1806.02169*, 2018.

[35] Xun Huang and Serge Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *The IEEE International Conference on Computer Vision (ICCV)*, Italy, Oct 2017.

[36] Mihir Parmar, Savan Doshi, Nirmesh J. Shah, Maitreya Patel, and Hemant A. Patil, "Effectiveness of cross-domain architectures for whisper-to-normal speech conversion," in *27th European Signal Processing Conference (EUSIPCO)*, Spain, 2019.

[37] Jaime Lorenzo-Trueba et al., "The Voice Conversion Challenge 2018: Promoting development of parallel and nonparallel methods," in *Speaker Odyssey*, France, 2018, pp. 192–202.

[38] D. Erro, I. Sainz, E. Navas, and I. Hernáez, "Improved HNM-based vocoder for statistical synthesizers," in *INTERSPEECH*, Florence, Italy, 2011, pp. 1809–1812.

[39] Laurens van der Maaten and Geoffrey Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[40] Hemant A. Patil, Madhu R. Kamble, Tanvina B. Patel, and Meet H. Soni, "Novel variable length Teager energy separation based instantaneous frequency features for replay detection.," in *INTERSPEECH*, 2017, pp. 12–16.

[41] Madhu R. Kamble, Hemlata Tak, and Hemant A. Patil, "Effectiveness of speech demodulation-based features for replay detection," in *Proceeding of INTERSPEECH*, 2018, pp. 641–645.

[42] Petros Maragos, Thomas F. Quatieri, and James F. Kaiser, "Speech nonlinearities, modulations, and energy operators," in *[Proceedings] ICASSP 91: 1991 International Conference on Acoustics, Speech, and Signal Processing*, Canada, 1991, IEEE, pp. 421–424.

[43] Madhu R Kamble and Hemant A Patil, "Novel amplitude weighted frequency modulation features for replay spoof detection," in *2018 11th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, China, 2018, IEEE, pp. 185–189.

[44] Stéphane Mallat, *A Wavelet Tour of Signal Processing*, Elsevier, 1999.

[45] Madhu R. Kamble and Hemant A. Patil, "Effectiveness of mel scale-based esa-ifcc features for classification of natural *vs.* spoofed speech," in *International Conference on Pattern Recognition and Machine Intelligence*. Springer, 2017, pp. 308–316.