Voice Style Transfer using StarGAN

Dhruuv Agarwal, Raunak Vijan, Satyaraja Dasara

Abstract—Voice style transfer can be defined as the transformation of voice characteristics like accent, pitch and tone from one speaker domain to another while retaining the linguistic content of the source speaker's speech in the transformed speech. Traditional approaches to this problem are heavily reliant on training with parallel data. However, parallel data containing same utterances by both the source and target speakers is scarce. The traditional approaches also tend to give over-smooth signal outputs and buzzy-sounding speech. Recent research using generative adversarial network architectures show potential in tackling this problem without the need for parallel data. These models have been shown to be successful in generating realistic sounding speech for the voice style transfer task. In this project we explore GAN architectures used for many to many voice style transfer,namely,CycleGAN and DiscoGAN. We implement StarGAN, a recent GAN architecture that requires less number of parameters compared to other traditional GAN architectures for this problem with an aim to produce similar results.

1 Introduction

The research problem that we seek to explore is voice style transfer. "Style transfer" has been an active area of research in the area of computer vision for the past couple of years. The same theory can be applied to the domain of speech where we aim to preserve the linguistic content of the signal while transforming the non-linguistic information in the signal. This problem can also be considered similar the well known voice conversion (VC) problem. The traditional method to approach this problem is to model it as a reconstruction task where the reconstruction is compared with the ground truth to optimize for the loss. Statistical and probabilistic approaches do not generalise well for such tasks because of the non-linear characteristic of this problem. Another shortcoming of this approach is the requirement of parallel data. This can be a major limitation due to low availability of such data. Additionally, such data can require considerable pre-processing such as time alignment procedures. This can prove detrimental to the actual learning task due to the realisation of such hard targets.

The recent advancement in the field of deep learning has shown great progress for the task of mapping style from one domain to another. Popular traditional deep learning based approaches include different types of variational auto encoders (VAE) [1]. These methods have still not mitigated the need of parallel data. The recent popularity of generative adversarial networks (GANs) [2]. in image style transfer, feature representation learning and image restoration tasks. These networks transform the problem from a mapping from one domain to another using parallel data into a discriminator based task between data that have the desired non linguistic qualities and those that don't. This forces the generator network to eventually learn to generate audio samples with a realistic mapping to the target speaker's speech domain. Another advantage of StarGAN [10]. is that it removes the limitation of one to one mapping between just one source and one target domain. It allows us to map from one to many or many to many domains. This can be attributed to the fact that the problem is now that of discrimination instead of domain to domain transformation.

2 Background and Related Work

2.1 CycleGAN [5]

This GAN architecture consists of two generators and two discriminators. It was traditionally used for one to one mapping without the need of parallel data. The cycle loss which is based on the reconstruction back to the speaker domain acts like a reinforcement to ensure the model learns the mapping in a bidirectional manner. Kaneko [5] used a variant of CycleGAN which used gated CNN and identity mapping loss. The generator consists of a downsampling block followed by six residual blocks further, followed by an upsampling block. The residual block has skip connections between the input and output of the block.The layers in the up-sampling and down-sampling blocks have no activation function. The upsampling and downsampling blocks have a gated linear unit (GLU) at the end to act as the activation for the block. Their model was compared with the baseline gaussian mixture model for voice conversion(GMM-VC). They evaluate the outputs on the basis of mean opinion scores and show that their reconstructions were more realistic.

2.2 DiscoGAN [3]

The high-level architecture of the DiscoGAN is very similar to that of the CycleGAN. The main difference between DiscoGAN and CycleGAN is that DiscoGAN has two reconstruction losses, one for each generator. This is in contrast to CycleGAN which uses just one cycle consistency loss to train both the generators. Gao [3]. used a modified version of DiscoGAN for a voice impersonation model. The proposed model has three discriminators instead of two. There are two discriminators for real or fake classification for each speaker domain. The third discriminator is to check if the source and the transformed signal are retaining the same style even if they belong to different voice domains for successful voice impersonation or mimicry task.

2.3 StarGAN

The initial work on StarGAN was for Style transfer between images Yunjey Choi et. al[]. This paper aimed to provide a

unified model to train one model for transfer between multiple domains. This was significant as it helped in improving the scaling and increasing the robustness of the model. Applying this approach to voice, we can limit trainable parameters if we seek to approach the problem of many to many voice style transfer. Kameoka et. al[5] adopted the StarGAN architecture given by Choi[4] for the task of many to many voice style transfer with non-parallel data. A significant reduction in test time inference was observed to allow for real time implementation with reasonably realistic voice quality.

3 PIPELINE

3.1 Preprocessing and Reconstruction

In order to extract acoustic features from the audio we use mel-cepstral coefficients. MFCC helps in extracting the components of the audio signal that are good for identifying the linguistic content and removes useless components like noise, emotions etc. MFCC features have been widely used in automatic speech and speaker recognition since they were introduced in 1980. MFCC consists of steps like fourier transform, mapping power spectrums to Mel scale and then taking discrete cosine transform of the Mel log powers. For our experiments we have computed MFCC from a spectral envelope obtained using pyworld library. We set the number of features to 36. After extracting MFCC features we get a spectrogram image of shape 36 x time. In order to reconstruct the signal we multiply the spectral envelope of input speech by the spectral gain function frame-by-frame and resynthesizing the signal using a vocoder.

3.2 Architecture and Model

Our work began with understanding and exploring CycleGAN[2] work on voice transfer between two domains. This helped in developing intuition about losses necessary in voice conversion tasks and tuning GAN's [10] in general. After seeing the impressive work by Kameoka et. al[5], we decided to base our work on this new concept. This approach of StarGAN helps in reducing the architecture to one generator and a two discriminators with shared weights. The is possible due to an auxiliary input label 'c' which tells the model the target domain to. This method makes sense considering the previous works like Conditional GANs[10] and conditional variational autoencoders (CVAEs)[1].

3.2.1 Generator

As discussed above, in StarGAN approach we have one Generator model handling multi domain transfers. To enable the transfer we take the input matrix image of the audio chunk and also the c(target domain) and c'(original domain) as inputs for the model. The domain information is first encoded as one hot vectors and then tiled to get their 3d representation. This is concatenated to the audio 'image' along the channel axis. This is passed to the Generator model to get the Generated audio image matrix in the target domain. This will be further used by discriminator to check for classifying domain(which speaker) and real/fake. This Generated fake audio image in target domain(speaker) is again sent to Generator after concatenating the original speaker information to ideally get back the original audio image. This reconstructed audio image is used to define Cyclic loss and identity losses for the generator model, as stated below.

$$\mathcal{L}_{\mathrm{cyc}}(G) = E_{c' \sim p(c), \mathbf{x} \sim p(\mathbf{x}|c'), c \sim p(c)} \left[\left\| G\left(G(\mathbf{x}, c), c'\right) - \mathbf{x} \right\|_{\rho} \right] \quad (1)$$

$$\mathcal{L}_{\mathrm{id}}(G) = E_{c' \sim p(c), \mathbf{x} \sim p(\mathbf{x}|c')} \left[\left\| G\left(\mathbf{x}, c'\right) - \mathbf{x} \right\|_{\rho} \right] \quad (2)$$

$$\mathcal{L}_{\mathrm{adv}}^{G}(G) = -E_{\mathbf{x} \sim p(\mathbf{x}), c \sim p(c)}[\log D(G(\mathbf{x}, c), c)] \quad (3)$$

The architecture of Generator consists of a U-Net[8] structure with encoder and decoder layers implemented using Convolutional and De-Convolutional layers. Between encoder and decoder section, we have put Residual layer[] blocks as bottleneck layers. The U-Net is commonly used in such applications as it retains the same size as input and can accommodate different sized inputs.

3.2.2 Discriminator

The Discriminator calculates loss on two classification tasks. It takes input as the real audio image of speaker and generated image from Generator. We have a network which is shared by the two classifiers. As the the two classification tasks are different we have different losses for both.

$$\mathcal{L}_{\mathrm{cls}}^{C}(C) = -E_{c \sim p(c), \mathbf{y} \sim p(\mathbf{y}|c)} \left[\log p_C(c|\mathbf{y}) \right] \quad (4)$$

$$\mathcal{L}_{\mathrm{cls}}^{G}(G) = -E_{\mathbf{x} \sim p(\mathbf{x}), c \sim p(c)} \left[\log p_C(c|G(\mathbf{x}, c)) \right] \quad (5)$$

$$\mathcal{L}_{\mathrm{adv}}^{D}(D) = -E_{c \sim p(c), \mathbf{y} \sim p(\mathbf{y}|c)}[\log D(\mathbf{y}, c)]$$
$$\text{-E}_{\mathbf{x} \sim p(\mathbf{x}), c \sim p(c)}[\log(1 - D(G(\mathbf{x}, c), c))] \quad (6)$$

For the architecture, we have convolutional layer blocks which downsamples the input, as the stride setting reduces size over time. Finally the last convolution layer is used to feed and calculate the two outputs, one for real/fake and one, softmax, for the domain/speaker classification.

To summarize, the full objectives of StarGAN-VC to be minimized with respect to G, D and C are given as

$$\mathcal{I}_G(G) = \mathcal{L}_{\mathrm{adv}}^{G}(G) + \lambda_{\mathrm{cls}}\mathcal{L}_{\mathrm{cls}}^{G}(G) + \lambda_{\mathrm{cyc}}\mathcal{L}_{\mathrm{cyc}(G)} + \lambda_{\mathrm{id}}\mathcal{L}_{\mathrm{id}}(G) \quad (7)$$

$$\mathcal{I}_D(D) = \mathcal{L}_{\mathrm{adv}}^{D}(D) \quad (8)$$

$$\mathcal{I}_C(C) = \mathcal{L}_{\mathrm{cls}}^{C}(C) \quad (9)$$

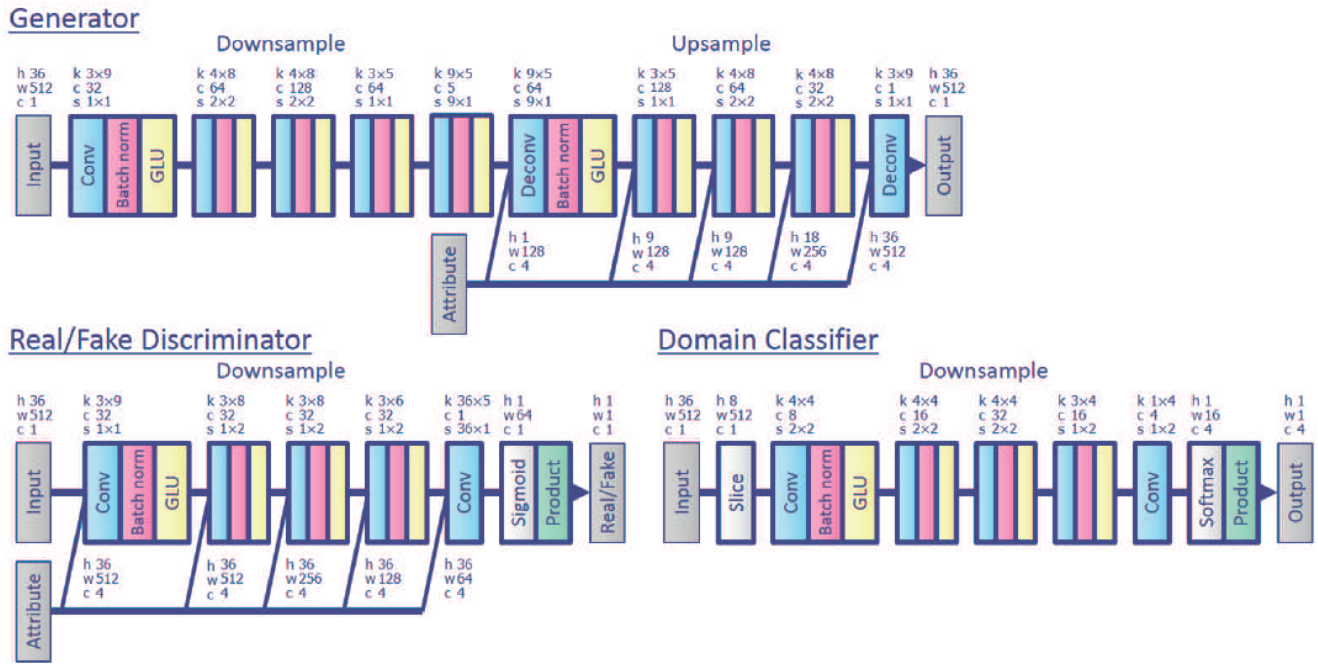

Fig. 1: Architecture of Model suggested in Paper

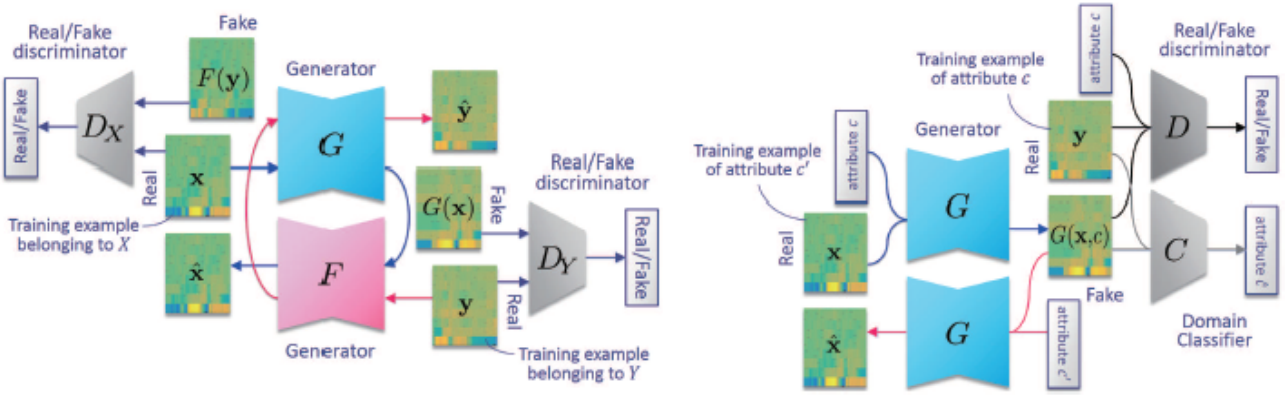

Fig. 2: Cycle GAN (left) vs Star GAN (right)

4 Datasets

CSTR VCTK[13] Corpus includes speech data uttered by 109 native speakers of English with various accents. Each speaker reads out about 400 sentences, most of which were selected from a newspaper plus the Rainbow Passage and an elicitation paragraph intended to identify the speaker's accent. We down-sampled the whole data from 48 kHz to 16 kHz,and operated on this data. Instead of training on the whole data which was of substantial size, we chose a small subset of 10 speakers as our chosen subset. The 10 speakers were selected from different region and of both male/female populations.The five regions were, 'Edinburgh', 'SouthEngland', 'Northern Irish Belfast', 'American New jersey' and 'India'. We then divided the audio files for these 10 speakers, as train and test files. Then we saved the the data in '.npy' and '.npz' files, and then later we pickled the data for quick loading.

5 Experiments

Following are the experiments we have done in the the model proposed by Kameoka et al[5] in original paper.

- For training the model, we have trained the generator once every 5 updates for discriminator update. This follows the suggestion in original GAN[] paper, which helps is balancing the Discriminator and Generator losses, so that either one doesn't perform too well.
- In every iteration we sample an audio image chunk of set dimensions of (36,256), where 36 is the Mel-cep

dimension and 256 is the set value, and speaker domain(which speaker). Thus our one iteration works a random sample of one speaker. Now for domain transfer, as paper suggests we sample a speaker from total number of speakers.

- The U Net structure, in each convolution blocks we have convolutional layer followed by instance normalization and then Rectified Linear Unit activation(ReLu). In discriminator we have used Leaky ReLu We have increased the number of bottleneck layers in the U Net structure, between the encoder and decoder sections to 6.
- We have used Adam optimizer with learning rate as 0.0001, beta 1=0.5 and beta 2= 0.999, for both generator and discriminator.
- While tuning the model we have experimented with the lambda values for the losses. Initially we set the lambda for cyclic loss as 10, but got better results for lambda =20. This was seen to better keep the linguistic content between the original and converted audio signals.
- We trained and tested on Google cloud with Nvidia Tesla P100 GPU.

6 Discussion

StarGAN seems to provide reasonably good domain transfer while maintaining the linguistic content for male-to-female and female-to-male voice conversion with same accent.

For male-to-female and female-to-male voice conversion between speakers having different accents the reconstruction is not that good but the domain transfer is reasonable. We see some patches of audio where the domain is transferred and some preservation of the linguistic content. However the audio seems to be noisy and fuzzy.

We observe that the results by StarGAN are not as impressive as the ones obtained in CycleGAN related work. This is because CycleGAN aims at only one-to-one mapping and hence it is intuitive that it would perform better on such a task. So, there seems to be a trade-off between quality of audio and generalization.

7 Conclusion and future work

This project allows non-parallel many-to-many voice conversion by using StarGAN. In comparison with CycleGAN which would require multiple generator for many-to-many mapping, StarGAN makes use of only one generator, thus it has only few parameter. The generator processes the signals quickly enough to support real-time applications.

As a future work we would like to train and evaluate our architecture on cross-language voice conversion by including other language like German, French, Spanish, Hindi, Chinese and Korean in addition to English.

At the moment the model is only capable to transfer voice on which it has been trained i.e. many-to-many voice conversion. We would like explore how we could handle any-to-any voice conversion that uses voice embeddings rather than a one-hot vector to identity the target and source speaker. This ambitious approach could allow for on-the-fly voice conversion between voices by using only short audio clip samples.

8 References

1) Diederik P Kingma, Max Welling.2013. Auto-Encoding Variational Bayes
2) Takuhiro Kaneko, Hirokazu Kameoka. 2017. Parallel-Data-Free Voice Conversion Using Cycle-Consistent Adversarial Networks
3) Yang Gao , Rita Singh, Bhiksha Raj. 2018. Voice impersonation using generative adversarial networks
4) Yunjey Choi , Minje Choi1, Munyoung Kim , Jung-Woo Ha ,Sunghun Kim , Jaegul Choo1. 2018. StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation
5) Hirokazu Kameoka, Takuhiro Kaneko, Kou Tanaka, Nobukatsu Hojo. 2018. StarGAN-VC: Non-parallel many-to-many voice conversion with star generative adversarial networks
6) Mehdi Mirza, Simon Osindero. 2014. Conditional generative adversarial nets
7) Chin-Cheng Hsu, Hsin-Te Hwang, Yi-Chiao Wu, Yu Tsao and Hsin-Min Wang. 2016. Voice Conversion from Non-parallel Corpora Using Variational Auto-encoder
8) Olaf Ronneberger, Philipp Fischer, Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation
9) Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. 2016. Deep Residual Learning for Image Recognition
10) Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio. 2014. Generative adversarial nets
11) Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, Jiwon Kim 2017. Learning to Discover Cross-Domain Relations with Generative Adversarial Network
12) Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks
13) VCTK Corpus. https://homepages.inf.ed.ac.uk.html