Conversion of Speech Using CNN

M.Mamatha
Assistant Professor, Dept of CSE
MGIT,Hederabad.

Abstract- *Traditional voice conversion methods rely on parallel recordings of multiple speakers pronouncing the same sentences. For real-world applications however, parallel data is rarely available. In this paper, demonstrating a voice conversion method that relies on non-parallel speech data and is able to convert audio signals of arbitrary length from a source voice to a target voice. We firstly compute spectrograms from waveform data and then perform a domain translation using Generative Adversarial Network (GAN) architecture. An additional Siamese network helps preserving speech information in the translation process, without sacrificing the ability to flexibly model the style of the target speaker. We test our framework with a dataset of clean speech recordings, as well as with a collection of noisy real-world speech examples. Finally, we apply the same method to perform music style transfer, translating arbitrarily long music samples from one genre to another, and showing that our framework is flexible and can be used for audio manipulation applications different from voice conversion.*

Keywords- Generative Adversarial Network (GAN), Siamese network, Spectrograms.

I. INTRODUCTION

We have all heard about image style transfer: extracting the style from a famous painting and applying it to another image is a task that has been achieved with a number of different methods. Generative Adversarial Networks (GANs in short) are also being used on images for generation, image-to-image translation and more. But what about sound? On the surface, we might think that audio is completely different from images, and that all the different techniques that have been explored for image-related tasks can't also be applied to sounds. But what if we could find a way to convert audio signals to image-like 2-dimensional representations?
This kind of sound representation is what we call "Spectrogram", and it is the key that will allow us to make use of algorithms specifically designed to work with images for our audio-related task. A spectrogram is a visual representation of the spectrum of frequencies of a signal as it varies with time. When applied to an audio signal, spectrograms are sometimes called sonographs, voiceprints, or voice grams.

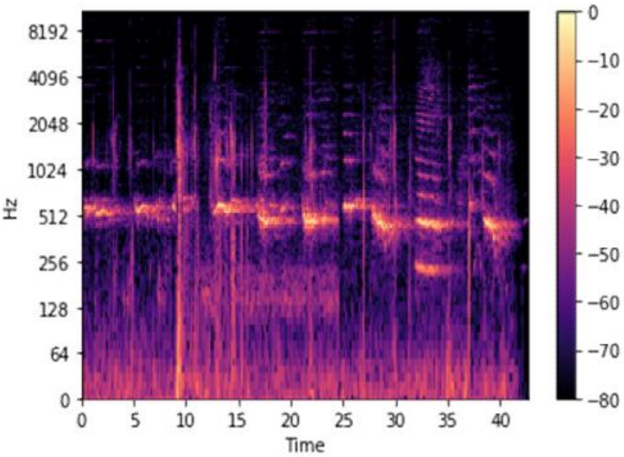

Fig 1 : Example of a Spectrogram

Given a time-domain signal (1 dimension) we want to obtain a time-frequency 2-dimensional representation. To achieve that, we apply the Short-Time Fourier Transform (STFT) with a window of a certain length on the audio signal, only considering the squared magnitude of the result. In simpler terms, we divide our original waveform signal into chunks that overlap with one another, extract the magnitude of the frequency in the chunk (with a Fourier Transform), and each resulting vector is going to represent a column of our final spectrogram. The x axis of the spectrogram stands for time, while the y axis represents the frequency.

To make these spectrograms even more useful for our task, we convert each "pixel" (or magnitude value) to be in the decibel scale, taking the log of each value. Finally, we convert spectrograms to the Mel scale, applying a Mel filter bank, resulting in what are known as "mel-spectrograms". This allows us to make the spectrogram representations more sensible to our human understanding of sound, highlighting the amplitudes and frequencies that us humans are more prone to hearing. It is also extremely important to note that spectrograms can be turned back into "audible" waveform data: it won't be a perfect reconstruction (phase information is missing in our magnitude spectrograms) but thanks to an algorithm called Griffin-Lim we are able to approximate phase and recreate realistically sounding audio.

Problem Statement:

We are to build and train a system capable of performing voice conversion and any other kind of audio style transfer (for example converting a music genre to another). The method is heavily inspired by recent research in image-to-image translation using Generative Adversarial Networks, with the main difference consisting in applying all these techniques to audio data. We will be able to translate samples of arbitrary length, which is something that we don't see very often in GAN systems.

Proposed System:

The above-mentioned system is possible to be built using the TraVeLGAN (Transformation Vector Learning GAN) GAN architecture. In addition to a Generator and a Discriminator (or Critic), TraVeLGAN introduces a Siamese network (a network that encodes images into latent vectors) to allow translations between substantially different domains keeping a content relationship between the original and converted samples.

II. METHODOLOGY

2.1 Choosing the Architecture:

There are a number of different architectures from the computer vision world that are used for image-to-image translation, which is the task that we want to achieve with our spectrogram representations of audio. Image-to-image translation consists in converting an image from a domain A (pictures of cats for example) to a different domain B (pictures of dogs), while keeping content information from the original picture (the expression and pose of the cat). Our task is practically the same: we want to translate from speaker A to speaker B, while keeping the same linguistic information from speaker A (the generated speech should contain the same words as the original speech from speaker A).

The most famous GAN architecture built for this goal may be Cycle GAN, introduced in 2017 and widely used since then. While Cycle GAN is very successful at translating between similar domains (similar shapes and contexts), such as from horses to zebras or from apples to oranges, it falls short when rained on very diverse domains, like from fishes to birds or from apples to carrots. The cause of this shortcoming is the fact that Cycle GAN heavily relies on pixel-wise losses, or in other words, its loss tends to minimize differences in pixel values of real and generated images: intuitively, when converting an image of an object (an apple for example) to a substantially different domain (carrot) we need to change the main shape of the original object, and Cycle GAN can't help us in this case. Spectrograms of speeches from different people (or spectrograms of musical pieces of different genres) can be very visually different from one another: thus, we need to find a more general approach to the problem, one that does not involve being constrained by translating between visually similar domains.

2.2 Proposed Solution – TraVeL GAN

Our goal is to find a way to keep a relationship between the original and generated samples without relying on pixel-wise losses (such as the cycle-consistency constraint in CycleGAN), that would limit translations between visually similar domains. Thus, if we encode the images (or spectrograms) into vectors that capture their content information in an organized latent space we are able to maintain a relationship between these vectors instead of the whole images.

That's exactly what a siamese network allows us to achieve. Originally used for the task of face recognition, the siamese network takes an image as input and outputs a single vector of length vec_len. Specifying with a loss function which image encodings should be close (images of the same face for example) in the vector space and which ones should be far apart (images of different faces) we are able to organize the latent space and make it useful for our goal.

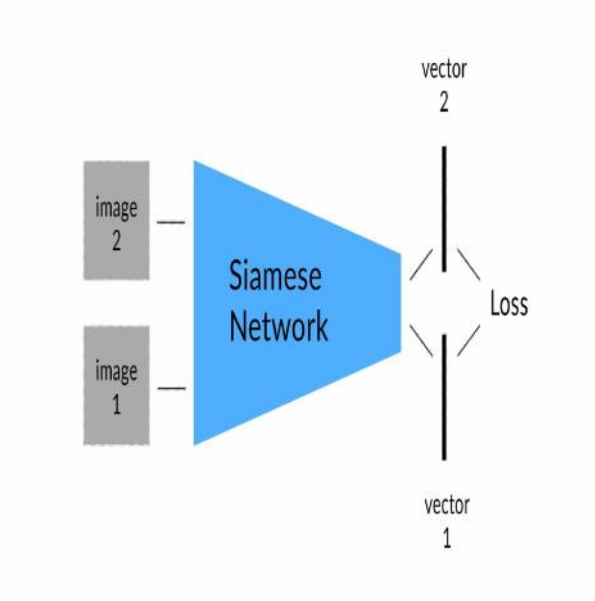

Fig 2: The Siamese network encodes images into vectors

More specifically, we aim at keeping the transformation vectors between pairs of encodings equal: this seems an extremely difficult concept to comprehend, but it is in fact quite easily understandable.

With G(X) as the translated image X (output of the generator), S(X) as the vector encoding of X and A1, A2 two images from the source domain A, the network must encode vectors such as:

(S(A1)-S(A2)) = (S(G(A1)-S(G(A2)))

In this way the transformation vector that connects encodings of a pair of source images must be equal to the transformation vector between the same pair translated by the generator.

This allows to preserve semantic information (differently from CycleGAN that preserves more geometric content information with its cycle-consistency constraint) in the translation, allowing the constraining of more "abstract" relationships between images of different domains.

Formally, to keep content information in the translation we will minimize the Euclidean distance and the cosine similarity between the two transformation vectors, so that both angle and magnitude of the vectors get preserved.

$$\mathcal{L}_{(G,S),TraVeL} = \mathbb{E}_{(a_{\frac{L}{2},1}, a_{\frac{L}{2},2}) \sim A}\left[cosine\_similarity(t_{12}, t'_{12}) + \|t_{12} - t'_{12}\|_2^2\right] \quad with\, a_{\frac{L}{2},1} \neq a_{\frac{L}{2},2}$$

$$t_{ij} = S(a_{\frac{L}{2},i}) - S(a_{\frac{L}{2},j})$$

$$t'_{ij} = S(G(a_{\frac{L}{2},i})) - S(G(a_{\frac{L}{2},j}))$$

The above is Formal TraVeL Loss.

Furthermore, it is important to clarify that both the generator and the siamese network must cooperate to achieve this objective. More specifically, the gradients of the TraVeL loss get back propagated through both of the networks and their weights get updated accordingly. Thus, while the discriminator and the generator have an adversarial objective (they challenge one another to reach their goal), the siamese and the generator help each other, cooperating under the same rules. In addition to this "content" loss, the generator will learn how to generate realistic samples thanks to a traditional adversarial loss

III. CONCLUSION AND FUTURE SCOPE

Conclusion:

We have seen how to perform voice translation and audio style transfer (such as music genre conversion) using a deep convolution neural network architecture and a couple of tricks and techniques to achieve realistic translations on arbitrarily long audio samples.

We now know that we are able to leverage a large part of the recent research on deep learning for computer vision applications to also solve tasks related to audio signals, thanks to the image-equivalent spectrogram representation.

Finally, I would like to conclude by acknowledging the fact that misusing this and other techniques for badly intentioned goals is possible, especially in the case of voice translation. With the rise of powerful machine learning methods to create realistic fake data we should all be very aware and cautious when exploring and using this kind of algorithms: and while the research won't stop and shouldn't be stopped, we should also allocate resources and look into how to detect the fake data that we helped creating.

Future Scope:

A good application of this project would be trying to create a dataset of voices from recordings of dead people and then creating an artificial intelligence trained to mimic the voices from the dataset using this algorithm. This also gives a chance for family members to hear the voices of their loved ones in their everyday life.

This helps reduce the stigma created by Hollywood movies and also makes it easier for AI to be more widely accepted by the common public.

REFERENCES

[1] Ian J. Goodfellow, Jean Pouget-Abadie et al., "Generative Adversarial Networks", 2014

[2] Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks", 2017

[3] Matthew Amodio, Smita Krishnaswamy, "TraVeLGAN: Image-to-image Translation by Transformation Vector Learning", 2019

[4] Tero Karras, Samuli Laine, Timo Aila, "A Style-Based Generator Architecture for Generative Adversarial Networks", 2019

[5] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, Jiwon Kim, "Learning to Discover Cross-Domain Relations with Generative Adversarial Networks", 2017

[6] Aaron van den Oord, Nal Kalchbrenner, Koray Kavukcuoglu, "Pixel Recurrent Neural Networks", 2016

[7] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks", 2016

[8] Mark French, Rod Handy, “Spectrograms: Turning Signals into Pictures”, 2007

[9] E.-J. Ong, S. Husain, and M. Bober, “Siamese network of deep fisher-vector descriptors for image retrieval”, 2017

[10] A. Royer, K. Bousmalis, S. Gouws, F. Bertsch, I. Moressi, F. Cole, and K. Murphy. Xgan, “Unsupervised image-to-image translation for many-to-many mappings”, 2017

[11] Z. Yi, H. R. Zhang, P. Tan, and M. Gong, “Dualgan: Unsupervised dual learning for image-to-image translation”, 2017