GAN for Audio Source Separation

Shaotian Chen
ECE Department
Schen121@ur.rochester.edu

Yueyi Yao
ECE Department
yyao25@ur.rochester.edu

Haikang Tan
AME Department
htan6@u.rochester.edu

ABSTRACT

Singing voice separation is a rising problem in the audio processing and machine learning area. The goal is to extract the voice track from the single original mixture audio. It may sound difficult, but a newly proposed machine learning model called Generative Adversarial Network (GAN) can be applied to provide a novel method for the problem. We regard the audio spectra as distributions and then use neuron network to process them. At first, we use ground truth (correct voice spectra) to initialize the parameters of generator network. The next step is to optimize them by a discriminator network until they converge. In this paper, we construct two GAN models to implement singing voice separation and compare the results with another two models: encoder-decoder and encoder-RNN-decoder. The results of experiments on dataset DSD100 show that the RNN-SVSGAN model has the best performance over other models.

Index Term—Singing voice separation, generative adversarial network, recurrent neural network, encoder-decoder, machine learning.

1. INTRODUCTION

An audio of a song is usually made up of voice track and instrument tracks (background music). However, when it comes to a monaural mixture audio, sometimes we only need the voice part, which can be really hard. This technique is called singing voice separation (SVS). It can be used to improve the effect of singing pitch estimation and cover song identification.

There are some traditional methods for SVS. One of the most widely used is non-negative matrix factorization (NMF). Also, it has been modified into some improved versions.

Since deep learning has been so popular these years, there are several approaches using neuron network. One of them is encoder-decoder, and the other one is encoder-RNN-decoder. The encoder-decoder architecture can estimate the vocal spectral mask from the mixture audio. Fig 1 shows a simple architecture of encoder-decoder for a single frame model.

The encoder-RNN-decoder method takes temporal information into consideration based on the former approach. It adds a long-short-time-memory (LSTM)

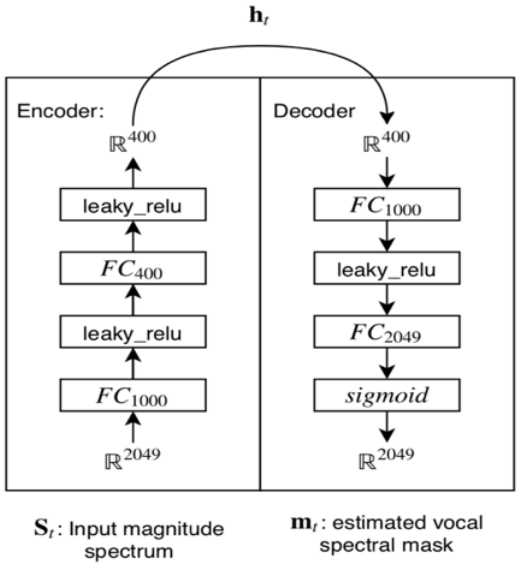

Figure 1. The encoder-decoder architecture for the single frame model. FC means the fully-connected layer (nn.Linear in Pytorch). Here we use FFT size 4096 and hop size 2048[3].

layer into the structure in order to summarize information across frames.

When it comes to the Generative Adversarial Network (GAN), it becomes quite popular recently. In the field of computer vision, it has achieved success in generating imitating fake images. It contains a generator network and a discriminator network which can use variable models. The goal of the generator is to try to generate fake samples which are close to the true ones, while the discriminator is going to judge whether the input is true. And the output of the generator will be the input of the discriminator. In other words, the generator tries to "cheat" the discriminator while the latter attempts to make the correct judgment. They compete with each other, and when the loss function obtains convergence, the output of the generator can be quite close to the realistic samples. In our experiment we will use a modified GAN called conditional GAN (cGAN) which takes mixture spectra into the input as well. With

the ground truth of voice spectra to initialize, the generator can be trained to generator a voice spectrum with inputting a mixture spectrum. So, it can be named as singing voice separation GAN (SVSGAN). The input samples of GAN will be the frames of audios after short-time Fourier Transform (STFT). Fig 2 shows the block diagram of the proposed framework. Fig 3 shows the flowchart of how GAN process audios.

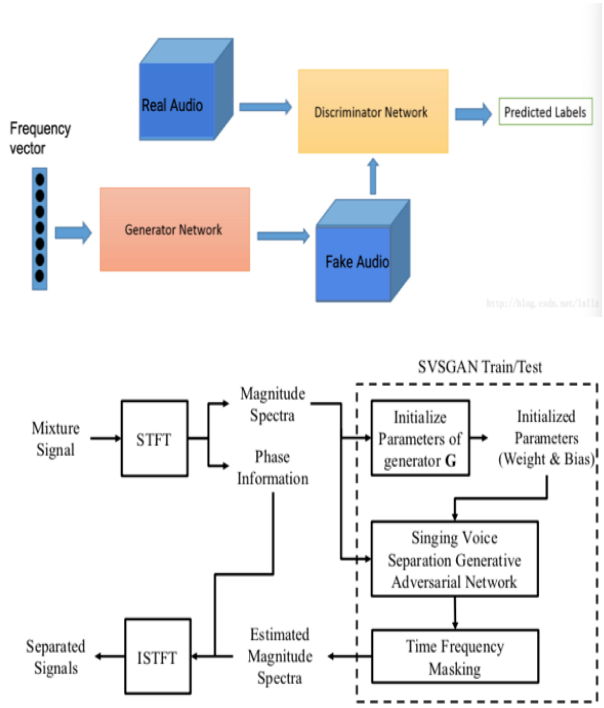

Figure 2. Block diagram of the proposed framework [2]

Figure 3. Working principle of GAN with audio processing

2. METHOD DESCRIPTION

Our figure is shown in Figure 2. Our method is divided into three stages above, which are stage of extraction of magnitude, stage of SVSGAN, stage of retrieval via ISTFT with phase reconstruction.

1.1 Network Selection

Generative adversarial networks (GANs) are deep neural net architectures comprised of two nets, pitting one against the other (thus the "adversarial"). GANs were first introduced by Ian Goodfellow and other researchers at the University of Montreal, including Yoshua Bengio, in 2014, GANs' potential is huge, because they can learn to mimic any distribution of data. That is, GANs can be taught to create worlds eerily similar to our own in any domain: images, music, speech, prose. They are robot artists in a sense, and their output is very impressive. So, in [2], people proposed the architecture of GAN like conditional GAN when applied to audio separation. Inspired by their work, we choose two better network structures to generate predicted voice spectra below, the first model is encoder and decoder in Fig 2, the second model is encoder and lstm and decoder[5] in Fig 4, where we add the lstm-rnn[6] between the encoder and decoder to apply the temporal information.

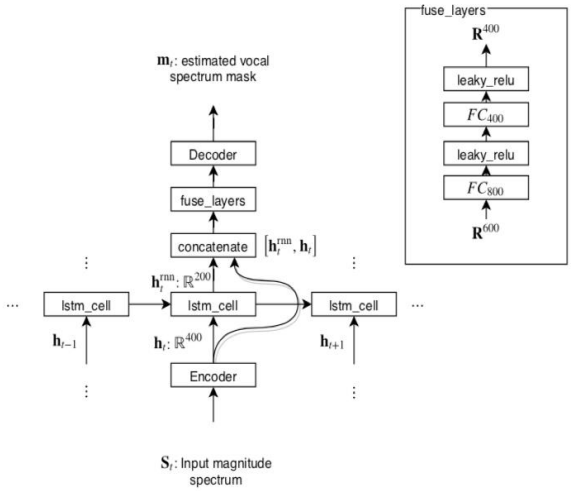

Figure 4. Encoder and lstm-rnn and decoder network structure for generator [3]

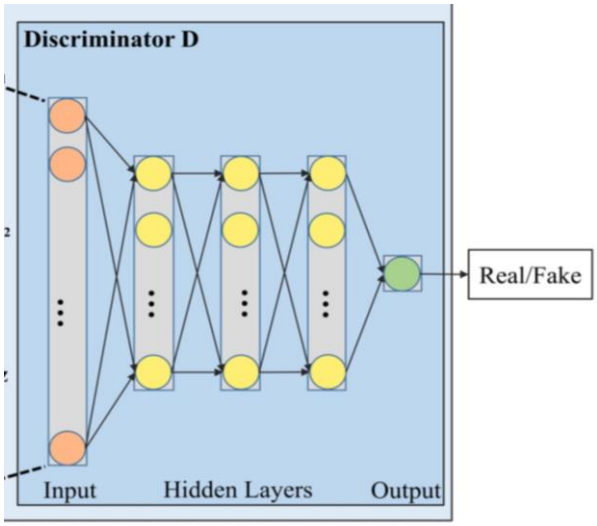

Figure 5. Discriminator D (tell us whether it is real) [2]

Then whole SVSGAN architecture consists of two conventional deep neural networks: Generator G and Discriminator D as shown in the Fig 5 below. First, we use magnitude spectra as features and take each spectrum as a sample vector from the spectra distribution. we perform nonlinear mapping between the input spectrum and the output spectrum. In our method, we only use the predicted voice spectrum as the output. Generator G inputs mixture spectra and generates realistic vocal part while Discriminator D distinguishes the clean spectra from those generated spectra. Indeed, what we get from the generator is a mask over the mixture of the spectrum, which corresponds the clean voice spectra. In this way, we could

calculate the spectrum of the voice using element wise multiplication, and then reconstruct the estimated singing voice using our estimated magnitude spectrum and the original mixture's phase with inverse STFT and overlap add method.

1.2 Pre-processing

In our model, during the process of data loader, we shuffle the frame index of the mixture when we are training the model in order to overcome the problem, and then take magnitude spectrogram via STFT (Short Time Fourier Transform) as input instead of image. In STFT, we use 2048 samples of hanning window with 1024 samples hop size to segment it into T frames.

1.3 Training Loss function

Before adversarial learning, the default loss function for this assignment is the generalized Kullback-Leibler divergence, as in the equation 1.

$$D(\boldsymbol{X}\|\boldsymbol{Y}) = \sum_{ij} x_{ij}\left(\log\left(x_{ij}+\epsilon\right) - \log\left(y_{ij}+\epsilon\right)\right) - x_{ij} + y_{ij} \quad (1)$$

Our goal for this generator is to minimize this distance between the estimated spectrum and the target spectrum, as in the equation 2.

$$\min \mathrm{E}[\mathrm{D}(v_t\|\widehat{v_t})] \quad (2)$$

where is a very small number, Matrix Notation is used here because we will deal with multiple instances batched together. The final value of the divergence should be averaged over batch size and the number of time steps in order to make the loss for different batch size and different songs comparable.

During adversarial training, our training objective function of GAN is defined as follows:

$$\min_G \max_D V_{SVSGAN}(G,D) = E_{x\sim P_{data(z,s_c)}}[logD(s_c,z)] + E_{z\sim P_{G(z)}}[\log(1-D(G(z),z)] \quad (3)[2]$$

Where Sc is just the clean voice spectra and the G(z) is the predicted voice spectra, which is generated from input spectra z. The output of discriminator is controlled by the input spectra z. From this step, our modal could not only learn the data distribution between the input mixture spectra and the output clean spectra. Here is our training process below.

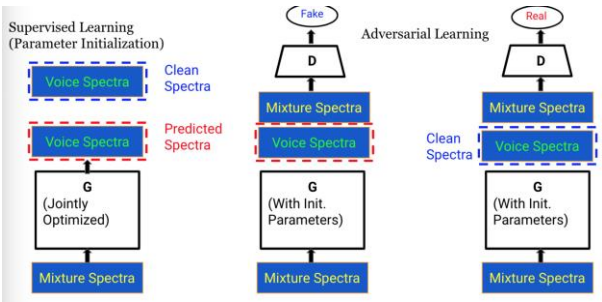

Figure 5. Training process of our whole modal [2]

First, we initialized the generator G with the parameters with Gaussian distribution, trained it in the supervised settings, and then put into the discriminator with the mixture of the spectra to get a label to tell us whether it is true or not.

1.4 *Phase reconstruction*

Since we only use magnitude spectrogram as input, we cannot get audio output directly. The phase information is unknown. Here, we use Griffin-Lim algorithm [4] to reconstruct phase

Algorithm 1 Griffin-Lim algorithm (GLA)

Fix the initial phase $\angle c_0$
Initialize $c_0 = s \cdot e^{i\angle c_0}$
Iterate for $n = 1, 2, \ldots$
$c_n = P_{\mathcal{C}_1}(P_{\mathcal{C}_2}(c_{n-1}))$
Until convergence
$x^* = \mathrm{G}^{\dagger} c_n$

This algorithm estimates phase information from magnitude information.

3. EXPERIMENT AND ANALYSIS

3.1 Dataset

The dataset we use is DSD100, it is a dataset of full lengths music tracks of different styles along with their isolated drums, bass, vocals and other stems. It is taken from a subtask called MUS from the signal separation evaluation campaign. The average duration of these songs is 4 minutes and 10 seconds. We separate it into three folder-train set, dev set, test set. We train on the train set, optimized in the dev set, and then test in the test set. We choose five different kinds of song to be our test set. Performance is measured in terms SDR, which is calculated by the blind source separation (BSS) Eval toolbox.

3.2 Comparison of 4 different modals

In fact, to compare the performance of the conventional encoder-decoder or encoder-lstm-decoder with two SVSGANs, their architectures of generator in them are

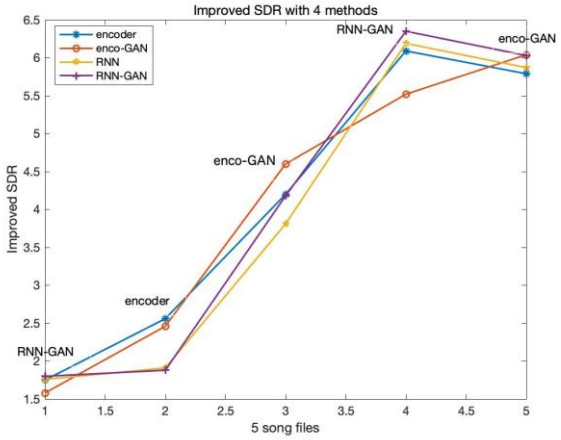

Figure 6. Signal Distortion Ratio of our different four modal.

identical to the encoder and decoder or encoder-lstm-decoder and are combined with discriminator D consisting 3 hidden layers, each with 512 neurons. The difference between them is the output of the generator in Fig.6 and Tab.1. We could clearly see that the modal using GAN enhance the performance of sound source separation in terms of SDR, encoder and decoder with GAN and encoder -lstm-decoder with GAN is found to achieve better results.

Signal Distortion Ratio (SDR)					
	Song1	Song2	Song3	Song4	Song5
Mixture	3.38	1.49	1.28	0.37	-2.20
Encoder	1.63	4.05	2.92	6.46	3.59
Encoder-GAN	1.80	3.95	3.32	5.89	3.84
RNN	1.62	3.40	2.53	6.56	3.67
RNN-GAN	1.58	3.37	2.90	6.72	3.83
Best SDR	1.80	2.56	4.60	6.35	6.04

Table 1. Signal Distortion Ratio of our different four modal.

3.3 Analysis and discussion

In the result, overall the modal with GAN perform better on the separation of different genres of songs. From this result, we could know that SVSGAN or SVSGAN using lstm-rnn could not only learn the mapping from the distribution of mixture spectra to the distribution of the clean spectra but also learn a general structure from the mixture structure at the same time. However, from our perspective, SVSGAN using lstm should achieve the best performance since they take temporal time information into consideration, but they do not outperform other models in all kinds of songs. First, we think our batch size are too small since we do not have the enough powerful machine to train this model. Despite of this, our model using GAN still achieve better results. For the next step, we will measure the average SDR over the whole test dataset and train it on an enough powerful machine using high batch size before testing.

4. FUTURE WORK

It seems the encoder-decoder with GAN method does not improve some kind of songs. We can try different optimizers and loss functions. Also, the batch size we used is 4, which may be too small to reach convergence. We can use 8 batches and allocate more memory for CPU in Google cloud.

In addition, in our experiment we reserve the phase of the original audio and add it to the result we obtain directly. In the future we would like to do the phase reconstruction via the phase decoder using reconstructed magnitude. There is an idea that we assume a von Mises distribution [7] for the phase and its derivatives. When the phase, the group delay and the instantaneous frequency are well satisfied, we can achieve a good phase estimation. In other words, the goal is to construct a joint model of the short-time Fourier transform magnitude spectra and phase spectrograms with a deep generative model.

5. REFERENCES

[1] Ian Goodfellow Generative adversarial nets. In Advances in neural information processing systems, pages 2672–2680, 2014.

[2] Zhe Cheng Fan Yen-Lin Lai SVSGAN: Singing voice separation via Generative adversarial network arXiv:1710.11428, 2017

[3] Y.Yan and Z.Duan: ECE477 computer audition HW5, 20118

[4] D. Griffin and J. Lim, “Signal estimation from modified short-time Fourier transform,” Acoustics, Speech and Signal Processing, IEEE Transactions on, vol. 32, no. 2, pp. 236 243, 1984.

[5] Antoine Liutkus, Fabian-Robert Stöter, Zafar Rafii, Daichi Kitamura, Bertrand Rivet, Nobutaka Ito, Nobutaka Ono, and Julie Fontecave. The 2016 signal separation evaluation campaign.

[6] Petr Tichavský, Massoud Babaie-Zadeh, Olivier J.J. Michel, and Nadège Thirion-Moreau, editors, Latent Variable Analysis and Signal Separation - 12th International Conference, LVA/ICA 2015

[7] Aditya Arie Nugraha, Kouhei Sekiguchi, Kazuyoshi Yoshii A deep generative model of speech complex spectrograms. https://arxiv.org/abs/1903.03269