Weakly Supervised Audio Source Separation
via Spectrum Energy Preserved Wasserstein Learning

Ning Zhang[12]**, Junchi Yan**[*1]**,Yuchen Zhou**[2]

[1] Shanghai Jiao Tong University, Shanghai, P.R. China
[2] IBM Research – China, Beijing, P.R. China
ningzh6610@gmail.com, yanjunchi@sjtu.edu.cn, zhouyuc@cn.ibm.com

Abstract

Separating audio mixtures into individual instrument tracks has been a standing challenge. We introduce a novel weakly supervised audio source separation approach based on deep adversarial learning. Specifically, our loss function adopts the Wasserstein distance which directly measures the distribution distance between the separated sources and the real sources for each individual source. Moreover, a global regularization term is added to fulfill the spectrum energy preservation property regardless separation. Unlike state-of-the-art weakly supervised models which often involve deliberately devised constraints or careful model selection, our approach need little prior model specification on the data, and can be straightforwardly learned in an end-to-end fashion. We show that the proposed method performs competitively on public benchmark against state-of-the-art weakly supervised methods.

1 Introduction

1.1 Background

Audio source separation (ASS) of mixed sources has long been a challenging problem, from which music source separation is a domain where considerable attentions have been attracted. Music source separation has demonstrated its tremendous potential values in various applications, such as music upmixing, audio restoration, music edit, music information retrieval, among others [Liutkus et al., 2015]. In fact, music source separation is challenging because it is an ill-posed inverse problem. Typically we have more signals to estimate than the number of signals we observed. As most music tracks are mono or stereo recordings, the task is to recover the sources of all its constituent instruments. This will be an underdetermined problem if no further regularization scheme is enforced.

Researchers have achieved successive breakthroughs on the music source separation techniques suited for specific

---

occasions. Among these techniques the guided source separation, which means algorithms employ some information about the audio sources or the acoustic mixing process, has shown promising performance for real-world practices. A few of these separation algorithms are conducted in spatial domain [Cardoso, 1998], but most methods switched to the time-frequency domain by means of complex-valued short time Fourier transform (STFT) [Vincent et al., 2014] or constant-Q transform (CQT) [Fitzgerald et al., 2011].

1.2 Related Work

Existing music source separation methods can be in general categorized into supervised approaches and unsupervised ones. For the supervised setting, the separated sources are assumed available as labeled data together with the mixture data. This direction has been recently dominated by neural network based methods. Due to the powerful modeling capability of the deep neural networks, these models achieve improvement in most audio source separation fields. [Uhlich et al., 2015] build a deep neural network (DNN) which takes magnitude spectrograms of mixture signal as input and tried to extract individual instruments from music. [Nugraha et al., 2016] propose to combine DNN with the classical Gaussian model to perform multichannel audio source separation. They further explore different distance metrics for the training of DNNs, including the Itakura-Saito divergence, Kullback-Leibler, Cauchy, mean squared error, and phase-sensitive cost functions. [Chandna et al., 2017] demonstrate the superiority of convolutional neural network (CNN) in monoaural audio source separation. [Huang et al., 2015] adopt a deep recurrent neural networks to separate monaural sources. [Uhlich et al., 2017] employ a data augmentation technique during training and blend an ensemble four DNNs and LSTM (long-short term memory) networks to do music audio source separation. Their blending model ranked the first place in the MUS task of the 2016 Signal Separation Evaluation Campaign [Liutkus et al., 2017].

The performance of these models largely depend on the assumption that there are large amount of labelled data for training, which is unrealistic in many applications. This restricts their applicability to real-world problems.

The unsupervised methods are mostly trying to solve the ill-posed inverse problem under different regularizations. The Independent Component Analysis (ICA) [Cardoso, 1998] or Sparse Component Analysis (SCA) [Comon and Jutten, 2010]
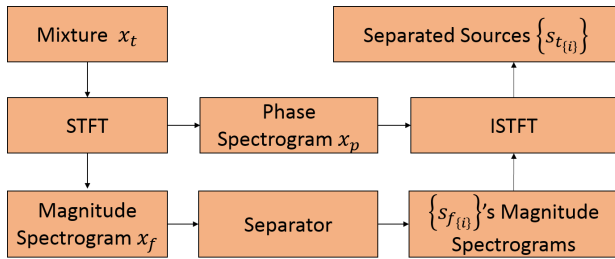
Figure 1: Working flow of music audio source separation.

based methods assume the source STFT coefficients have a stationary non-gaussian distribution [Vincent, 2007]. The local Gaussian modelling (LGM) based methods [Liutkus *et al.*, 2011] assumed that the vectors of STFT coefficients of the source spatial images have a zero-mean nonstationary Gaussian distribution. Compared to totally unsupervised separation, combining different level of weakly supervised information about the sources leads to improved performance in practice. The non-negative matrix factorization (NMF) based methods [Ozerov and Fevotte, 2010] employ the global structure information of sources and demonstrated impressive performance. The REPET algorithm [Rafii and Pardo, 2013] takes advantage of the prior that most musical background is repetitive whereas the vocal signal is not.

These unsupervised and weakly supervised methods are built on specific and adhoc assumptions, which limit their generalization capability. Besides, these methods require expensive computation and high processing time, making them difficult to use for real-world applications. Last but not least, state-of-the-art unsupervised models, which in essence are weakly supervised models, e.g. kernel additive model (KAM) [Liutkus *et al.*, 2014] and OZE [Ozerov *et al.*, 2012] all heavily rely on a precise and correct prior knowledge on the separated sources, the deliberate constraint design and kernel selection prevent them from user-friendly. More importantly, in many real-world applications, such prior knowledge is either hard to collect or difficult to be converted to a mathematical formula.

1.3 Approach Overview

This paper aims to explore weakly supervised methods as seeing their potential for real-world applications. We pursue a more general approach aiming at relying less on constraints, assumptions, prior knowledge on the source data as done by previous methods [Liutkus *et al.*, 2014; Ozerov *et al.*, 2012]. Specifically our model is based on two mild assumptions making them inherently more general: i) the signal's spectrum energy is preserved after separation which has been a well accepted concept in literature [Sandsten, 2016]; ii) the separated sources are in the same distribution with other source data, e.g. separated drum sources from music shall lie in its own distribution which can be approximated by a few (other) real-world drum samples – which are more easier to collect than in the supervised case as we require no one-to-one correspondence between the separated source and its mixture. In fact, the above two constraints can be complementary to each other as the first one enforces global consistency while

the latter pays more attention to its own distribution with no interlock to other separated sources in learning.

From the optimization perspective, the second assumption above can be mathematically converted to the Wasserstein distance which has been proven robust metric to measure the distance between two distributions [Arjovsky *et al.*, 2017]. In contrast to the KL-divergence that requires strict match between two probability distributions and does not consider how close two samples are but only their relatively probability, which is sensitive to sample noise and outliers, Wasserstein distance is more sensitive to the underlying geometry structure of samples and more robust to noise/outliers. In addition, the first assumption can be explicitly accounted by an analytical energy preservation term in the loss function. To make the Wasserstein distance more tractable, in this paper we turn to the generative adversarial nets (GANs) [Goodfellow *et al.*, 2014] technique. More specifically, the Wasserstein distance minimization problem can be solved by a minmax gaming procedure if we treat our separation model as the generator and introduce an additional discriminator that tries to distinguish the fake generated sources from the real ones. As a result, the efforts boil down to devising an effective generator (see Figure 3) and discriminator (see the discriminator modules in Figure 2). For the discriminator structure, skip connection is used in this paper to improve its capacity. More importantly, we employ n independent discriminators to account for their corresponding sources respectively. The global consistency is synergically complemented by the energy preservation term. In very recent preprint works, there are also GAN based source separation networks. The SVSGAN in [Fan *et al.*, 2017] turns to a supervised approach whereby GAN technique is additionally used as an enhancement. Another work [Subakan and Smaragdis, 2017] turns to unconditional GAN for unsupervised learning of different separated tracks, for the purpose of directly learning their different distributions using Wasserstein distance. In contrast, our model is based on the conditional GAN which can directly handle input source for separation, and our end-to-end learning algorithm is totally different from their piece-by-piece ad-hoc method.

To better formulate the problem, we assume the number of sources n for separation is fixed and known, and we further assume all the training and testing data are associated with a same n for separation. This assumption is realistic for real-world applications and has been widely adopted by existing works on music source separation [Vincent *et al.*, 2014].

1.4 Contribution

In a nutshell, the highlights of this paper are:

1) We explore a new paradigm (a similar concurrent idea appears in [Stoller *et al.*, 2018]) for weakly supervised source separation, which involves a deep network learned by a general loss function which rely less on prior knowledge on each of the source data for separation, compared with existing weakly supervised models [Liutkus *et al.*, 2014; Ozerov *et al.*, 2012]. The direct benefit is for its end-to-end learning capability.

2) We present a novel and general loss involving i) the Wasserstein distance between the distributions of the separated source samples and the real-world source samples; ii) the spectrum energy preservation constraint which can be a

complementary to each source's Wasserstein loss, as it imposes global interlock and consistency among the separated sources compared with the mixed signal. We explore the joint use of these two complementary losses to address the highly under-determined problem.

3) Our learning paradigm and loss function, lead to competitive performance on music separation benchmark compared with state-of-the-art weakly supervised methods. In contrast to [Liutkus *et al.*, 2014] needing careful kernel model selection, and [Ozerov *et al.*, 2012] calling for deliberate constraint design, our model is easy to implement and more general.

2 Weakly Supervised Wasserstein Learning for Audio Source Separation (ASS)

2.1 Preliminaries on ASS

We explain the energy preserved Wasserstein learning for separating the mixture audio x_t into sources $\{s_{t_i}\}_{i=1...n}$. Here n denotes the number of sources. The overall data flow is shown in Figure 1. Firstly, the mixture audio x_t is segmented into overlapped segments of time context C, on which the short time Fourier transform (STFT) is computed. The resulting magnitude spectrograms $x_f(\omega)$ are passed through the separator, which outputs the estimate $\{\hat{s}_{f_i}(\omega)\}_{i=1...n}$ for each of the separated sources i. Here ω is the time indices. These estimates, along with the computed phase of the mixture, are transformed through an inverse STFT to recover the audio signals $\hat{x}_t$ corresponding to the separated sources.

2.2 Energy Preserved Wasserstein Learning

As mentioned in the introduction, the loss function involves i) the energy preservation term to restrict the separated sources's total energy is close to the mixed one; ii) the distribution distance term hoping the separated sources are similar to other real-world sources in the same category.

Spectrum Energy Preservation

This constraint is widely accepted and used in signal processing literature e.g. [Sandsten, 2016], which can be written by:

$$\sum_{i=1}^{n} \|\hat{s}_{f_i}(\omega)\|_2^2 = \|x_f(\omega)\|_2^2,$$

Wasserstein Distance between Distributions

We describe the Wasserstein distance in the language of GANs. It involves two losses L_g, L_d for generator and discriminator respectively, as written by [Goodfellow *et al.*, 2014]:

$$
\begin{cases}
L_d = & -\mathbb{E}_{\mathcal{S}_f \sim \mathbb{P}(\mathcal{S}_f)}(D_\theta(\mathcal{S}_f)) + \mathbb{E}_{x_f(\omega) \sim \mathbb{P}(x_f)}(D_\theta(G_w(x_f(\omega)))) \\
& + \lambda \mathbb{E}_{\tilde{\mathcal{S}}_f(\omega) \sim \mathbb{P}(\tilde{\mathcal{S}}_f(\omega))} \left[\left(\|\nabla_{\tilde{\mathcal{S}}(\omega)} D_\theta(\tilde{\mathcal{S}}_f(\omega))\|_2 - 1 \right)^2 \right] \\
L_g = & -\mathbb{E}_{x_f(\omega) \sim \mathbb{P}(x_f)} \left(D \left(G_w \left(x_f(\omega) \right) \right) \right) \\
& + \lambda \mathbb{E}_{\tilde{\mathcal{S}}_f(\omega) \sim \mathbb{P}(\tilde{\mathcal{S}}_f(\omega))} \left[\left(\|\nabla_{\tilde{\mathcal{S}}_f(\omega)} D_\theta(\tilde{\mathcal{S}}_f(\omega))\|_2 - 1 \right)^2 \right]
\end{cases}
\tag{1}
$$

where θ and w are the model parameter for the generator and discriminator, respectively. While $\mathcal{S}_f = \{s_{f_1}, ..., s_{f_n}\}$ denotes n sources, and λ is gradient penalty weight. $\tilde{\mathcal{S}}(\omega) \sim \mathbb{P}(\tilde{\mathcal{S}}(\omega))$ are points sampled uniformly along straight lines
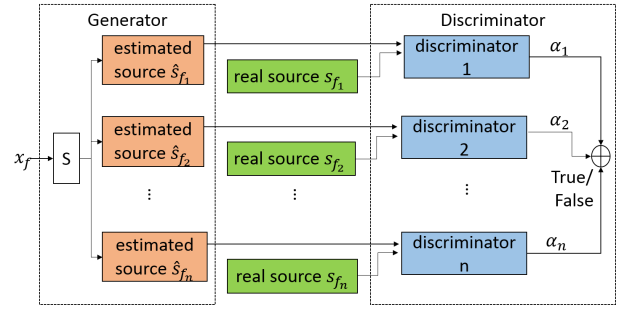

Figure 2: Overview of the proposed framework for source separation. The input mixture signal is first transformed into the spectrum domain as x_f, then it is separated into n sources $\hat{s}_{f_i}$. For each source, a discriminator is devised by using the separated i.e. generated sources as well as the real sources from auxiliary source data. The overall discriminative loss is a combination of multiple sources.

between pairs of points sampled from the data distribution $\mathbb{P}(\mathcal{S}(\omega))$ and the estimated distribution $\mathbb{P}(\hat{\mathcal{S}}(\omega))$. Refer to [Gulrajani *et al.*, 2017] for the detailed justifications.

Energy Preservation Wasserstein Loss

By combing the above two loss functions, our objective function can be further written as a minmax problem whose objectives can be optimized alternatively:

$$
\min_\theta \max_w \underbrace{\mathbb{E}_{\mathcal{S}_f \sim \mathbb{P}(\mathcal{S}_f)}(D_\theta(\mathcal{S}_f)) - \mathbb{E}_{x_f(\omega) \sim \mathbb{P}(x_f)}(D_\theta(G_w(x_f(\omega))))}_{\text{Wasserstein loss between separated and real source}}
$$
$$
\underbrace{- \lambda \mathbb{E}_{\tilde{\mathcal{S}}_f(\omega) \sim \mathbb{P}(\tilde{\mathcal{S}}_f(\omega))} \left[\left(\|\nabla_{\tilde{\mathcal{S}}(\omega)} D_\theta(\tilde{\mathcal{S}}_f(\omega))\|_2 - 1 \right)^2 \right]}_{\text{1-Lipschitz regularizer}}
$$
$$
\underbrace{+ \beta \left(\sum_{i=1}^{n} \|\hat{s}_{f_i}(\omega)\|_2^2 - \|x_f(\omega)\|_2^2 \right)^2}_{\text{Energy preservation}}
\tag{2}
$$

where β is the energy integrity penalty weight. Here we impose a Lipschitz restriction similar to [Gulrajani *et al.*, 2017], which is denoted as $\|f\|_L \leqslant K$, for $K \geqslant 0$ – for all real x_1 and x_2, $|f(x_1) - f(x_2)| \leqslant K|x_1 - x_2|$.

Note that existing audio separation models often use additional regularization terms to maximize the independence between the estimated sources. In this paper, we do not use the independence regularization explicitly. Because the independent restriction is implied in the above generative adversarial process as the estimated sources are forced to be sampled from real source distributions. This helps simplify our model.

In training stage, we randomly select mixture signals and input them to the generator G_θ. The estimated sources for these mixture will then be output. These estimates, together with randomly selected real sources, are then feed into the discriminator D_w. Note that each kind of source was input to a specific individual discriminator d_w for each source. We then alternately optimize D_w and G_θ through gradient decent on the losses of them. After training, the generator can be taken as a music source separator.
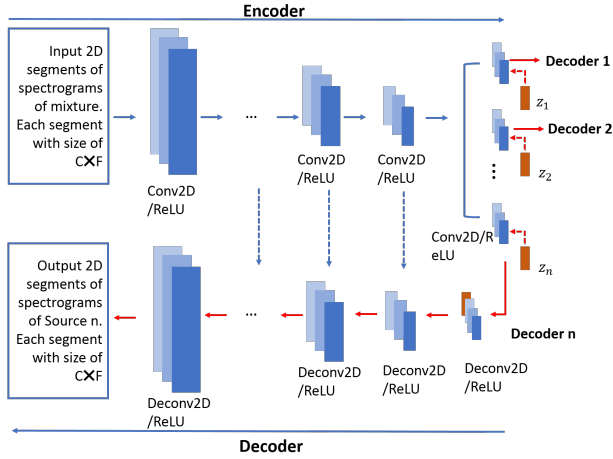
Encoder

Decoder

Figure 3: Structure of the generator (only the decoder for source n is depicted). The decoders for other sources share the similar structure to decoder n. 'conv2D/ReLU' denotes a 2D convolutional layer followed by rectified linear unit (ReLU). 'deconv2D/ReLU' denotes a 2D convolutional layer followed by rectified linear unit (ReLU). C denotes the time context and F denotes the frequency bins. z_i is the random noises added to each output of the encoder.

Algorithm 1 Source Separation GAN (SSGAN): Energy preserved Wasserstein learning of audio source separation.

Require: Sectrograms of mixture audio $\{x_f(\cdot)\}$, and the spectrograms of source audios $s_{f_i(\cdot)}, i = 1, ..., n$

Require: $G_\theta(\cdot)$: generator; $D_w(\cdot)$: discriminator; β: energy integrity penalty weight; λ: gradient penalty coefficient; m: batch size; I_{critic}: number of discriminator iterations per generator iteration. ω: time indice.

1: Initialize $G_\theta(\cdot)$ and $D_w(\cdot)$ with random weights.
2: **repeat**
3:     Randomly select m spectrograms of mixture audios $\{x_f(\omega)\}, \omega = 1, ..., m$. Input them to the generator G_w. Get n estimated sources for each spectrogram $\{\hat{s}_{f_i}(\omega)\}, \omega = 1, ..., m; \ i = 1, ..., n$.
4:     Update generator parameters θ via optimizing Equation 2 by Adam [Kingma and Ba, 2014].
5:     **for** d-steps in I_{critic} **do**
6:         Randomly select m spectrograms of each real source $\{s_{f_i}(\omega)\} \sim \mathbb{P}_{s_{f_i}}, \omega = 1, ..., m; i = 1, ..., n$. Randomly select m spectrograms of mixture data $\{x_f(\omega)\} \sim \mathbb{P}_x, \omega = 1, ..., m$ as input to D_θ
7:         Update discriminator parameters w via gradient decent by Equation 2 by Adam [Kingma and Ba, 2014].
8:     **end for**
9: **until** model converges

The training procedure of the proposed generative adversarial separation model is described in Algorithm 1.

2.3 Architecture of ASS Generator/Discriminator

In our model, the generator aims to separate the mixed signal into n sources, while the discriminator tries to distinguish each separated source with the real ones. We use two neural networks to model both of the generator and discriminator. The working flow is illustrated in Figure 2.

For the generator i.e. separator, we propose to use a structure of convolutional auto-encoder with one channel of input and n channels for output. The network takes single channel mixture $x_f(\omega)$ as input, and outputs n estimated signals $\{\hat{s}_{f_i}(\omega)\}_{i=1...n}$ with n channels. Each channel corresponds to a kind of source signal. The separator is depicted in Figure 3, which is a fully convolutional auto-encoder. In the encoding stage, the input mixture spectrograms are mapped into low dimensional features by a number of strided convolutional layers followed by rectified linear units (ReLUs). For decoding, these low dimensional features are projected back to high dimensional outputs via a number of fractional strided convolutional layers. No dense layer is used. We devise this architecture after the guidelines in [Radford *et al.*, 2015].

The encoding part of the generator aims to extract low dimensional source-specific features from the input data. Hence the encoder is devised to extract source-common features through several convolutional layers and then output n source-specific features in the last layer. Before passing these source-specific features to decoders, we add some noise to increase diversity to the output of separator. Note that the generator network consists of skip connections [Ronneberger *et al.*, 2015], connecting the encoding layers to its corresponding decoding layers. These skip connections can pass low level details directly to higher layers, and help the decoder to reconstruct the source signals. Additionally, gradients in back propagation process can flow deeper via the skip connections.

The discriminator $D(\cdot)$ is another CNN which is larger and deeper than generator. It takes in the estimated source signals and output a similarity score between the distribution of separated sources $\mathbb{P}(\hat{s}_f)$ and that of the real sources $\mathbb{P}(s_f)$. For modeling flexibility, we devise separate discriminators $d_i(\cdot)$ for each source, and then combined their losses together to obtain the final loss: $D(\{s_1, ..., s_n\}) = \sum_i^n \alpha_i * d_i(s_i)$, $d_i(\cdot)$ where α_i is its corresponding weight.

3 Experiments and Discussion

3.1 Evaluation Protocol

Dataset We term our method as source separation GAN (SS-GAN). We convert the stereo songs into mono by computing the average of the two channels for all songs and sources in Demixing Secrets Dataset (DSD100)[1], which is designed to evaluate the performances of music audio source separation methods. Dataset consists of 100 professionally produced full track songs from the Mixing Secrets Free Multitrack Download Library[2]. For each song, the mixture and the sources have the same length and are all sampled at 44.1 K Hz. The DSD100 is divided into a dev set and a test set. Each of them consist of 50 songs. We train our proposed model on the dev set and test the trained model on the test set. The magnitude spectrogram for the data is computed by STFT. We use non-overlapping Hanning window [Podder *et al.*, 2014] of 1024

---

[1]http://liutkus.net/DSD100.zip

[2]http://cambridge-mt.com/ms-mtk.htm

parameter	description	value
C	time context	32
F	frequency bins	513
α_1	weight of discriminator of bass	0.25
α_2	weight of discriminator of drum	0.25
α_3	weight of discriminator of vocal	0.4
α_4	weight of discriminator of others	0.1
β	energy integrity penalty weight in Eq. 1	10
λ	gradient penalty weight in Eq. 1	10

Table 1: Parameter notations and settings.

	layer	# filters	output shape
Encoder	conv2D(3,3,2)/ReLU	16	(16,256,16)
	conv2D(3,3,2)/ReLU	32	(8,128,32)
	conv2D(3,3,2)/ReLU	64	(4,64,64)
	conv2D(3,3,2)/ReLU	128	(2,32,128)
	conv2D(3,3,2)	256*4	(1,16,256)*4
	noise(1,16,1)/ReLU	—	(1,16,256)*4
Decoder	deconv2D(3,3,2)/ReLU	128	(2,32,128)/Skip1:
	deconv2D(3,3,2)/ReLU	64	(4,64,64)/Skip2:
	deconv2D(3,3,2)/ReLU	64	(8,128,32)/Skip3:
	deconv2D(3,3,2)/ReLU	64	(16,256,16)
	deconv2D(3,3,2)/ReLU	1	(32,513,1)

Table 2: The detailed structure of generator (i.e. separator). The output shape is shown as (time-context, frequency bins). 'conv2D(3,3,2)' denotes 2D convolutional layer with filter size 3×3, and stride = 2 in both time-context direction and frequency bin direction. 'LeakyReLu(0.2)' denotes a leaky rectified linear layer with slide 0.2. 'Skip' denotes a skip connection from the corresponding encoder layer. The input data is of size 32 frames and 513 frequency bins. All the decoders share the same structure

layer	# filters	output shape
conv2D(3,3,2)/ LeakyReLU(0.2)	64	(16,256,64)
conv2D(3,3,2)/ LeakyReLU(0.2)	64	(8,128,64)
conv2D(3,3,1)/ LeakyReLU(0.2)	128	(8,128,128)
conv2D(3,3,2)/ LeakyReLU(0.2)	128	(4,64,128)
conv2D(3,3,2)/ LeakyReLU(0.2)	256	(2,32,256)
conv2D(3,3,1)/ LeakyReLU(0.2)	256	(2,32,256)
FC layer/ LeakyReLU(0.2)	—	(1)

Table 3: Detailed structure of each discriminator. The output shape is shown as (time-context, frequency bins). 'conv2D(3,3,2)' denotes 2D convolutional layer with filter size 3×3, and stride = 2 in both time-context direction and frequency bin direction. 'LeakyReLu(0.2)' denotes a leaky rectified linear layer with slide 0.2. The input data is of size $32 * 513$, denoting 32 frames and 513 frequency bins.

points, which corresponds to about 23 milliseconds. For each frame, the first 513 FFT points were taken as magnitude values. For the input and output data for the SSGAN, we aggregate magnitude vectors in a context window of 32 frames. Hence, one SSGAN input vector has a size of 32×513, and each input and output instant spans around 743 milliseconds. All of the inputs are scaled to $[0, 1]$.

This dataset also provides source tracks for *drums*, *vocals*, *bass*, and *other* instruments for each song in the set. we set generator with four output channels, and the discriminator with four input channels. Each of the channels corresponds to a kind of source. Details of the generator and discriminator networks can be found in Table 2 and Table 3. The generator is composed of an encoder and four decoders. The encoder consists 5 convolutional layers followed by rectified linear units. The last layer of the encoder has four output channels, to each a random noise is added. The four decoders are of the same structure. Each step in the decoder consists of a fractional strided convolutional layers followed by rectified linear units, and a concatenation with the correspondingly encoder layer. The four discriminators share the same structure. Each discriminator contains 6 convolutional layers followed by leaky rectified linear units, and a fully connected (FC) output layer indicating the similarity between the estimated source and the real source. The outputs from the four discriminators are weighted summed together into the final score.

Model settings The parameters for the generator and discriminator networks were initialized randomly. We used the Adam optimizer [Kingma and Ba, 2014] with hyperparameters $\alpha = 0.0001, \beta_1 = 0.5, \beta_2 = 0.9$ to train the generator and the discriminator, using a batch size of 16.

The other parameters related to the model setup can be found in Table 1. We note that the *other* source comprises many kind of instruments, and varied a lot across songs. Besides, some of the segments in the *other* source is closed to the *vocals* source. This indicates that the distribution of the *other* source is highly dispersed, and is sometimes confounded with that of the *vocals*. To learn the distribution of the *other* source well is more difficult than that of *drums*, *vocals*, and *bass*. Hence we reduce the weight of the discriminator for the *other* , and raise the weight for *vocals*.

Metrics Following the widely used protocol, we measure the performance of the proposed separation method by the source to distortion ratio (SDR), source to interference ratio (SIR), source to artifacts ratio (SAR), source image to spatial (ISR). Among them, SDR is usually considered as the overall

performance for source separation [Vincent *et al.*, 2006].

We compute the above metrics using BSS Eval toolbox [Vincent *et al.*, 2006]. The SiSEC 2016 (Signal Separation Evaluation Campaign) published the evaluation results of all the submitted methods[3]. This facilitates the comparison of our SSGAN based method with others. We note that the published results did not cover all the 50 songs in the test set. Only the results of 47 songs are published. To make it fair, we also compute our results on the same 47 songs in the test set.

Compared weakly supervised methods We choose three state-of-the-art weakly supervised methods: kernel additive modellings (KAM1 and KAM2) [Liutkus *et al.*, 2014; 2015], and OZE (as termed by the SiSEC 2016 – Signal Separation Evaluation Campaign https://sisec.inria.fr/) [Ozerov *et al.*, 2012] for comparison as our method is also weakly supervised. Note that the three peer methods are not based on deep network. In fact we have not identified any weakly supervised source separation deep network during the production of this

---

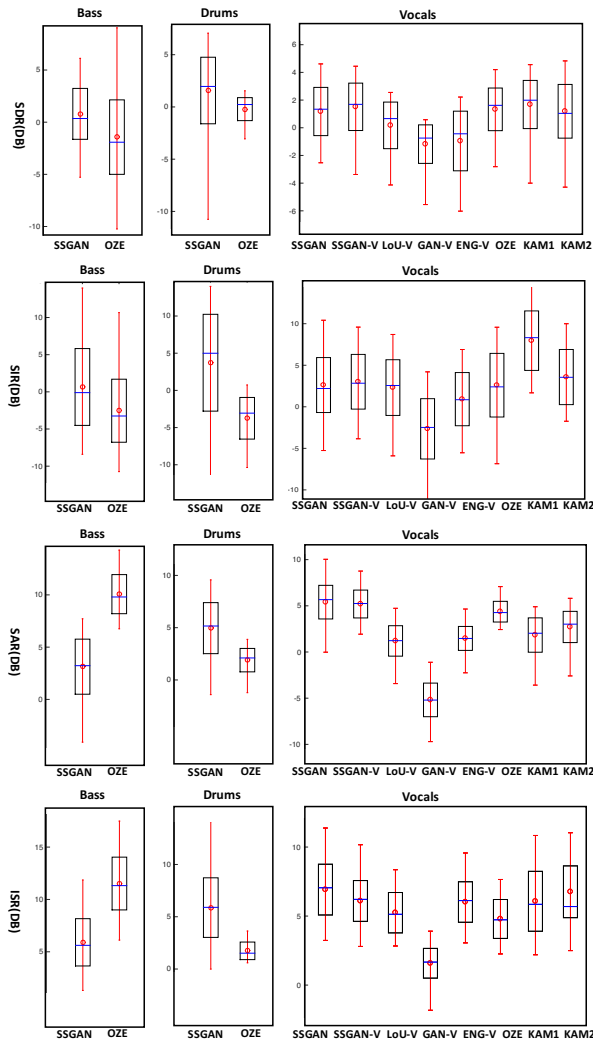[3]https://sisec17.audiolabs-erlangen.de/#/methods

Figure 4: The overall performance (SDR) and the three individual metrics (SIR, SAR, ISR) for *bass*, *drums*, *vocals* separation task and the *vocals* vs. *non-vocals* separation task on the test part of DSD100. The ends of red vertical lines denote the maximum and minimum values, and the red ∘ denote the mean values. The upper and lower edges of the boxes denote the $mean \pm standard\ deviation$. The horizontal line in the box denotes the median values.

paper, until a concurrent work in [Stoller *et al.*, 2018].

3.2 Results and Discussion

The performance of peer methods are all from SiSEC Competition 2016 results[4]. As the peer methods KAM1 and KAM2 [Liutkus *et al.*, 2015] only disclose performance on *vocals* vs. *non-vocals* separation task, hence we also test this case and denote as SSGAN-V (for *vocals*) for our method. The full separation result and the binary *vocals* vs. *non-vocals* separation are both shown in Figure 4. For our approach, the full separation and two-source separation models are separately trained by setting different number of sources and discriminators.

---

[4]https://sisec17.audiolabs-erlangen.de/#/results/1/4/2

To prove the superiority of the proposed model. We also test the effect of the U-net structure. For the *vocals* vs. *non-vocals* separation task, we test the performance when the proposed model lacking of U-net structure, denoted as LoU-V. Besides, to verify the complementary effect of the energy-preserving loss and the wasserstein loss, we test the performance of the *vocals* vs. *non-vocals* separation task when the energy-preserving loss is removed, denoted as GAN-V, and when only the energy-preserving loss is used, denoted as ENG-V. These experimental results are shown in Figure 4.

Our model performs competitively against the peer methods. In particular, SSGAN's overall performance SDR outperforms on *bass* and *drums*. The average SDR improvements of the proposed methods over OZE method is approximately 1.32 dB. A comparison test using a non-parametric Mann-Whitney U test, at the 5% confidence interval level, shows that the proposed method is significantly better than the OZE method on the *drum*and *bass* channels, while it shows that the difference on the *vocals* channel is not significant. We also note our binary separation version SSGAN-V slightly improves the full version SSGAN on the *vocals* vs. *non-vocals* separation task in terms of the overall performance SDR. For SDR on *vocals*, the performance difference is not significant. We conjecture this is because learning *vocals*'s real data's distribution is more challenging to learning its Wasserstein distance as the *vocals* can contain diverse patterns.

When the U-net connections are removed, the performances of LoU-V dropped obviously compared to that of SSGAN-V. This demonstrates the effect of the U-net connections. The performances of GAN-V and ENG-V are significantly lower than that of SSGAN-V. This proves our idea that the wasserstein loss and the energy-preserving loss are complementary. The energy-preserving loss can constrain the multiple sources as a whole while the wasserstein loss can constrain each source separately to make each of them look more realistic. There is quite obviously synergy between the two losses.

It is important to note that the OZE method [Ozerov *et al.*, 2012] leverages auxiliary RWC dataset [Goto *et al.*, 2003] which contains a large amount of independent *bass* and *drums* source data to pretrain their model. Comparatively we only use the relatively small data from DSD100 dev (50 songs), which may cause the difference in performance. The KAM model [Liutkus *et al.*, 2014] also requires deliberate model selection to fit with the source data which is no need in our method. Without such fine-tuning, the performance of KAM may drop as discussed in their paper [Liutkus *et al.*, 2014].

4 Conclusion

This paper is an endeavor to explore weakly supervised neural network methods for source separation, which is a relatively less studied compared with the dominant supervised learning based network models. As source separation is an ill-posed problem we resort to adversarial learning which has not been well explored for source separation in literature. We devise a loss involving spectrum energy preservation term and Wasserstein loss between the separated sources and the real source data. Our model is easy to implement and free from deliberate design of ad-hoc constraints and assumptions on the sources, thus being promising for practical applications.

References

[Arjovsky *et al.*, 2017] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

[Cardoso, 1998] J Cardoso. Multidimensional independent component analysis. In *ICASSP*, pages 1941–1944, 1998.

[Chandna *et al.*, 2017] Pritish Chandna, Marius Miron, Jordi Janer, and Emilia Gómez. Monoaural audio source separation using deep convolutional neural networks. In *International Conference on Latent Variable Analysis and Signal Separation*, pages 258–266. Springer, 2017.

[Comon and Jutten, 2010] Pierre Comon and C. Jutten. *Handbook of Blind Source Separation: Independent Component Analysis and Separation*. 2010.

[Fan *et al.*, 2017] Zhe-Cheng Fan, Yen-Lin Lai, and Jyh-Shing R. Jang. Svsgan: Singing voice separation via generative adversarial network. *arXiv preprint arXiv:1710.11428*, 2017.

[Fitzgerald *et al.*, 2011] Derry Fitzgerald, Rajesh Jaiswal, Eugene Coyle, and Scott Rickard. Shifted nmf using an efficient constant-q transform for monaural sound source separation. *Dublin Institute of Technology*, 2011.

[Goodfellow *et al.*, 2014] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014.

[Goto *et al.*, 2003] Masataka Goto, Hiroki Hashiguchi, Takuichi Nishimura, and Ryuichi Oka. Rwc music database: Music genre database and musical instrument sound database. *Ipsj Sig Notes*, 3:19–26, 2003.

[Gulrajani *et al.*, 2017] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans. *arXiv preprint arXiv:1704.00028*, 2017.

[Huang *et al.*, 2015] Po-Sen Huang, Minje Kim, Mark Hasegawa-Johnson, and Paris Smaragdis. Joint optimization of masks and deep recurrent neural networks for monaural source separation. *IEEE/ACM Transactions on Audio, Speech and Language Processing*, 23(12):2136–2147, 2015.

[Kingma and Ba, 2014] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *Computer Science*, 2014.

[Liutkus *et al.*, 2011] Antoine Liutkus, Roland Badeau, and Gel Richard. Gaussian processes for underdetermined source separation. *IEEE Transactions on Signal Processing*, 59(7):3155–3167, 2011.

[Liutkus *et al.*, 2014] Antoine Liutkus, Derry Fitzgerald, Zafar Rafii, Bryan Pardo, and Laurent Daudet. Kernel additive models for source separation. *IEEE Transactions on Signal Processing*, 62(16):4298–4310, 2014.

[Liutkus *et al.*, 2015] Antoine Liutkus, Derry Fitzgerald, and Zafar Rafii. Scalable audio separation with light kernel additive modelling. In *ICASSP*, pages 76–80, 2015.

[Liutkus *et al.*, 2017] Antoine Liutkus, Fabian-Robert Stöter, Zafar Rafii, Daichi Kitamura, Bertrand Rivet, Nobutaka Ito, Nobutaka Ono, and Julie Fontecave. The 2016 signal separation evaluation campaign. In *International Conference on Latent Variable Analysis and Signal Separation*, pages 323–332. Springer, 2017.

[Nugraha *et al.*, 2016] Aditya Arie Nugraha, Antoine Liutkus, and Emmanuel Vincent. Multichannel audio source separation with deep neural networks. *IEEE Transactions on Audio, Speech, and Language Processing*, 24(9):1652–1664, 2016.

[Ozerov and Fevotte, 2010] Alexey Ozerov and Cric Fevotte. Multi-channel nonnegative matrix factorization in convolutive mixtures for audio source separation. *IEEE Transactions on Audio Speech and Language Processing*, 18(3):550–563, 2010.

[Ozerov *et al.*, 2012] Alexey Ozerov, Emmanuel Vincent, and Frédéric Bimbot. A general flexible framework for the handling of prior information in audio source separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(4):1118–1133, 2012.

[Podder *et al.*, 2014] Prajoy Podder, Tanvir Zaman Khan, Mamdudul Haque Khan, and M Muktadir Rahman. Comparative performance analysis of hamming, hanning and blackman window. *International Journal of Computer Applications*, 96(18):1–7, 2014.

[Radford *et al.*, 2015] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *Computer Science*, 2015.

[Rafii and Pardo, 2013] Zafar Rafii and Bryan Pardo. Repeating pattern extraction technique (repet): A simple method for music/voice separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(1), 2013.

[Ronneberger *et al.*, 2015] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.

[Sandsten, 2016] Maria Sandsten. Time-frequency analysis of time-varying signals and non-stationary processes. *Lund University, Centre for Mathematical Sciences, available online at http://www. maths. lth. se/matstat/kurser/masm26/2016/TIMEFREQkompendie*, 2016.

[Stoller *et al.*, 2018] Daniel Stoller, Sebastian Ewert, and Simon Dixon. Adversarial semi-supervised audio source separation applied to singing voice extraction. In *ICASSP*. IEEE, 2018.

[Subakan and Smaragdis, 2017] Y.Cem Subakan and Paris Smaragdis. Generative adversarial source separation. *arXiv preprint arXiv:1710.10779*, 2017.

[Uhlich *et al.*, 2015] Stefan Uhlich, Franck Giron, and Yuki Mitsufuji. Deep neural network based instrument extraction from music. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 2135–2139. IEEE, 2015.

[Uhlich *et al.*, 2017] Stefan Uhlich, Marcello Porcu, Franck Giron, Michael Enenkl, Thomas Kemp, Naoya Takahashi, and Yuki Mitsufuji. Improving music source separation based on deep neural networks through data augmentation and network blending. *ICASSP*, 2017.

[Vincent *et al.*, 2006] E. Vincent, R. Gribonval, and C. Fevotte. Performance measurement in blind audio source separation. *IEEE Transactions on Audio Speech and Language Processing*, 14(4):1462–1469, 2006.

[Vincent *et al.*, 2014] Emmanuel Vincent, Nancy Bertin, Remi Gribonval, and Frederic Bimbot. From blind to guided audio source separation: How models and side information can improve the separation of sound. *IEEE Signal Processing Magazine*, 31(3):107–115, 2014.

[Vincent, 2007] Emmanuel Vincent. Complex nonconvex l p norm minimization for underdetermined source separation. In *International Conference on Independent Component Analysis and Signal Separation*, pages 430–437, 2007.