Crossfire Conditional Generative Adversarial Networks for Singing Voice Extraction

Weitao Yuan[1], *Shengbei Wang*[1*], *Xiangrui Li*[1], *Masashi Unoki*[2], *Wenwu Wang*[3]

[1]Tianjin Key Laboratory of Autonomous Intelligence Technology and Systems, Tiangong University, China
[2]Japan Advanced Institute of Science and Technology, Nomi, Ishikawa, Japan
[3]Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford, UK

weitaoyuan@hotmail.com, wangshengbei@tiangong.edu.cn, darkforever@outlook.com, unoki@jaist.ac.jp, w.wang@surrey.ac.uk

Abstract

Generative adversarial networks (GANs) and Conditional GANs (cGANs) have recently been applied for singing voice extraction (SVE), since they can accurately model the vocal distributions and effectively utilize a large amount of unlabelled datasets. However, current GANs/cGANs based SVE frameworks have no explicit mechanism to eliminate the mutual interferences between different sources. In this work, we introduce a novel 'crossfire' criterion into GANs to complement its standard adversarial training, which forms a dual-objective GANs, namely Crossfire GANs (Cr-GANs). In addition, we design a Generalized Projection Method (GPM) for cGANs based frameworks to extract more effective conditional information for SVE. Using the proposed GPM, we extend our Cr-GANs to conditional version, i.e., Crossfire Conditional GANs (Cr-cGANs). The proposed methods were evaluated on the DSD100 and CCMixter datasets. The numerical results have shown that the 'crossfire' criterion and GPM are beneficial to each other and considerably improve the separation performance of existing GANs/cGANs based SVE methods.

Index Terms: generative adversarial networks, crossfire criterion, generalized projection method, singing voice extraction

1. Introduction

Separating the singing voice from the accompaniment in music recordings, namely Singing Voice Extraction (SVE), is a challenging task [1], since the vocal component and accompaniment are highly correlated in both time and frequency [2]. Extensive studies have been conducted to address this problem, which can be organized into model-based methods and data-driven methods [2]. Compared with model-based methods [3–7], the data-driven approaches, notably the deep learning based methods [8], have provided a strong boost to the SVE performance [9–15].

Many deep learning methods for SVE use hand-crafted loss or distribution functions, which may lead to biased models [16]. In contrast, Generative Adversarial Networks (GANs) [17] which employ *adversarial* training between the generator and discriminators, do not need to specify the output distributions, thus provide potentially more accurate modeling [16,18–20]. In addition, different from most supervised SVE methods which require labelled datasets [2], the *adversarial* training of GANs can work in a semi-supervised fashion [20] by making use of unlabelled data. As a variant of GANs, Conditional GANs (cGANs), which utilize conditional information (CI) to improve the discriminator, have also been used for SVE [18, 19].

In spite of this progress, there are two main issues in current GANs/cGANs based SVE frameworks. First, the SVE aims to separate different sources from the mixture; thus, it is preferred to eliminate the mutual interference between vocal and accompaniment. However, the standard GANs/cGANs have no explicit mechanism to handle this issue. Second, the estimated sources depend heavily on the input mixture, while the current cGANs for SVE simply feed the mixture (i.e., the CI) into the discriminator via a naive concatenation of the mixture and the real or estimated (faked) sources [18,19]. According to [17,21], the optimal conditional discriminator for cGANs is the sum of two log likelihood ratios (see Eq. (2) in [21]), in case of SVE, the first ratio should model the relationships between the CI (the mixture) and the real or estimated (faked) sources, and the second ratio, which is independent of CI, should model the intrinsic characteristics for each source. However, the simple concatenation in current SVE methods cannot fully comply with this architecture. How to effectively use CI to design a better SVE discriminator needs to be further investigated.

To address the first issue, this paper introduces a novel criterion (loss), i.e., a *'crossfire'* criterion, to the traditional *adversarial* training of GANs. The 'crossfire' criterion aims to mutually eliminate the interferences between different sources. This loss and the traditional adversarial loss form a novel dual-objective GANs, namely Crossfire GANs (Cr-GANs). To address the second issue, the Generalized Projection Method (GPM), inspired from the Projection Method (PM) [21], is proposed for cGANs. In GPM, effective CI and intrinsic features are extracted by mapping the mixture and the real/faked source respectively to the same low-dimensional high-level feature space, and then the correlation between CI and intrinsic features is computed to improve the conditional discriminator. Accordingly, the GPM can fit the architecture of cGANs better. Using the proposed GPM, we extend our Cr-GANs to Crossfire Conditional GANs (Cr-cGANs) so that the proposed 'crossfire' criterion and GPM can work together for SVE. To fairly compare the proposed method with the state-of-the-art GANs based SVE method [20], we implement the proposed Cr-GANs and Cr-cGANs in the same framework as [20] and compare their performance under the same conditions.

2. Crossfire GANs (Cr-GANs)

The proposed Cr-GANs for two-source separation is illustrated in Fig. 1, where $\mathbb{X}_1$ ($\mathbb{X}_2$) is the source space with the distribution p_{data_1} (p_{data_2}) and $\mathbb{Y}$ is the mixture space with distribution p_{mix}. The generator $\boldsymbol{G}(y) = (G_1(y), G_2(y))$, $y \in \mathbb{Y}$, is the separator, where $G_1 : \mathbb{Y} \to \mathbb{X}_1$ and $G_2 : \mathbb{Y} \to \mathbb{X}_2$. The dis-

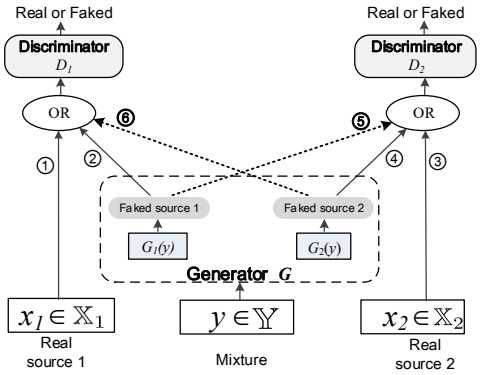

Figure 1: *The proposed Cr-GANs, where dotted lines ⑤ and ⑥ represent the 'crossfire' criterion.*

criminator $D_1 : \mathbb{X}_1(\cup\mathbb{X}_2) \to \mathbb{R}$ and $D_2 : \mathbb{X}_2(\cup\mathbb{X}_1) \to \mathbb{R}$ aim to distinguish which is 'true' or 'faked' for each source. For 'crossfire' criterion, each discriminator is fed with the opposite (interfering) faked source (see lines ⑤ and ⑥ in Fig. 1). We use two types of training datasets: labelled and unlabelled. The labelled dataset is $\mathcal{D}_{\text{p}} = \{(\mathbf{s}_1, m_1), \ldots, (\mathbf{s}_M, m_M)\}$, including M pairs of the mixture $m_i \in \mathbb{Y}$ and its sources $\mathbf{s}_i = (s_i^1, s_i^2), s_i^1 \in \mathbb{X}_1, s_i^2 \in \mathbb{X}_2, 1 \le i \le M$. The unlabelled datasets are the mixture dataset $\mathcal{D}_{\text{u}} = \{m_1^{\text{u}}, \ldots, m_U^{\text{u}}\} \subset \mathbb{Y}$ and the solo source datasets $\mathcal{D}_{\text{s}}^k \subset \mathbb{X}_k, k = 1, 2$, where U is the number of unlabelled mixtures, and the mixtures and sources in unlabelled datasets are not paired.

The training of Cr-GANs in Fig. 1 includes two alternating steps: (i) training the generator by fixing the discriminators and (ii) training the discriminators by fixing the generator. We adopted the same loss function as in [20] to train the generator:

$$\min_{\boldsymbol{G}} \quad L_{\text{s}}(\boldsymbol{G}) + \alpha L_{\text{u}}(\boldsymbol{G}) + \beta L_{\text{add}}(\boldsymbol{G}), \tag{1}$$

where $L_{\text{s}}(\boldsymbol{G})$ is the supervised loss, $L_{\text{u}}(\boldsymbol{G})$ is the unsupervised loss, and $L_{\text{add}}(\boldsymbol{G})$ is the additive penalty. The α controls the influence from the adversarial (or crossfire) discriminators and β controls the weight of the additive penalty. The defination of $L_{\text{s}}(\boldsymbol{G})$, $L_{\text{u}}(\boldsymbol{G})$, and $L_{\text{add}}(\boldsymbol{G})$ are

$$L_{\text{s}}(\boldsymbol{G}) = \frac{1}{M}\sum\nolimits_{i=1}^{M} ||\boldsymbol{G}(m_i) - \mathbf{s}_i||_2, \text{where } (\mathbf{s}_i, m_i) \in \mathcal{D}_{\text{p}}, \tag{2}$$

$$L_{\text{u}}(\boldsymbol{G}) = -\mathbb{E}_y[D_1(G_1(y))] - \mathbb{E}_y[D_2(G_2(y))], \tag{3}$$

$$L_{\text{add}}(\boldsymbol{G}) = \frac{1}{U}\sum\nolimits_{i=1}^{U} ||\sum_{k=1}^{2} G_k(m_i^{\text{u}}) - m_i^{\text{u}}||_2, m_i^{\text{u}} \in \mathcal{D}_{\text{u}}. \tag{4}$$

The discriminators are trained with the gradient penalty of the improved Wasserstein-GAN [22] to alleviate mode collapses [23],

$$L_k^{\text{grad}}(x) = \mathbb{E}_x \max(||\nabla_x D_k(x)||_2 - 1, 0)^2, k = 1, 2, \tag{5}$$

where x involves randomly interpolating between the real data and the output of generator (see Table 1). The L_k^{grad} aims at constraining the gradient norm of each discriminator's output with respect to its input [20] [22].

2.1. Traditional Adversarial Loss

The traditional adversarial loss, as defined in Eq. (6), only aims at increasing the recognition ability of each discriminator for its own real source,

$$\min_{D_1, D_2} \quad -W_p(D_1, D_2) + \lambda P_{\text{adv}}(D_1, D_2), \tag{6}$$

Table 1: *The different sampling points for gradient penalty.*

Type	Discriminator	Sampling point	One end	The other end
Adversarial	D_1	$\hat{x}_1^{\text{adv}}$	x_1	$G_1(y)$
	D_2	$\hat{x}_2^{\text{adv}}$	x_2	$G_2(y)$
Crossfire	D_1	$\hat{x}_1^{\text{cross}}$	x_1	$G_2(y)$
	D_2	$\hat{x}_2^{\text{cross}}$	x_2	$G_1(y)$

where the Wasserstein distance $W_p(D_1, D_2)$ and the penalty $P_{\text{adv}}(D_1, D_2)$ are defined as

$$\begin{aligned} W_p(D_1, D_2) &= \underbrace{\mathbb{E}_{x_1 \sim p_{\text{data}_1}}[D_1(x_1)]}_{\text{solid line ① in Fig. 1}} - \underbrace{\mathbb{E}_{y \sim p_{\text{mix}}}[D_1(G_1(y))]}_{\text{solid line ② in Fig. 1}} \\ &\quad + \underbrace{\mathbb{E}_{x_2 \sim p_{\text{data}_2}}[D_2(x_2)]}_{\text{solid line ③ in Fig. 1}} - \underbrace{\mathbb{E}_{y \sim p_{\text{mix}}}[D_2(G_2(y))]}_{\text{solid line ④ in Fig. 1}}, \end{aligned} \tag{7}$$

$$P_{\text{adv}}(D_1, D_2) = L_1^{\text{grad}}(\hat{x}_1^{\text{adv}}) + L_2^{\text{grad}}(\hat{x}_2^{\text{adv}}), \tag{8}$$

where x_1 (x_2) in Eq. (7) is sampled from $\mathcal{D}_{\text{s}}^1$ ($\mathcal{D}_{\text{s}}^2$), y is sampled from $\mathcal{D}_{\text{u}}$, and $\hat{x}_k^{\text{adv}} (k = 1, 2)$ in Eq. (8) is sampled in a standard way [22] (see row 2-3 in Table 1). When alternately solving Eqs. (6) and (1), the traditional GANs can be obtained.

2.2. Crossfire Criterion

With the traditional adversarial loss, discriminators can only concentrate on distinguishing their own sources (real or faked), but know nothing about the interfering source. To enhance it, we introduce the following crossfire criterion,

$$\min_{D_1, D_2} \quad -V_c(D_1, D_2) + \lambda P_{\text{cr}}(D_1, D_2), \tag{9}$$

where the crossfire distance $V_c(D_1, D_2)$ and the penalty $P_{\text{cr}}(D_1, D_2)$ are defined as

$$\begin{aligned} V_c(D_1, D_2) &= \underbrace{\mathbb{E}_{x_1 \sim p_{\text{data}_1}}[D_1(x_1)]}_{\text{solid line ① in Fig. 1}} - \underbrace{\mathbb{E}_{y \sim p_{\text{mix}}}[D_1(G_2(y))]}_{\text{dotted line ⑥ in Fig. 1}} \\ &\quad + \underbrace{\mathbb{E}_{x_2 \sim p_{\text{data}_2}}[D_2(x_2)]}_{\text{solid line ③ in Fig. 1}} - \underbrace{\mathbb{E}_{y \sim p_{\text{mix}}}[D_2(G_1(y))]}_{\text{dotted line ⑤ in Fig. 1}}, \end{aligned} \tag{10}$$

$$P_{\text{cr}}(D_1, D_2) = L_1^{\text{grad}}(\hat{x}_1^{\text{cross}}) + L_2^{\text{grad}}(\hat{x}_2^{\text{cross}}), \tag{11}$$

where the sampling points $\hat{x}_k^{\text{cross}} (k = 1, 2)$ in Eq. (11) work in a crossfire way (see row 4-5 in Table 1).

Using the crossfire distance V_c, each discriminator is fed with its real source and the interfering 'faked' source from the generator. Owing to this, two discriminators have the chance to see the interferences from the interfering source so that the interferences can be discovered and eliminated. Therefore, the crossfire criterion, which complements the standard adversarial loss, improves the ability of each discriminator in discriminating against the interfering source and prevent the two output distributions of the generator from corrupting each other.

3. Crossfire Conditional GANs (Cr-cGANs)

3.1. Implementation of Cr-cGANs

To further improve the discriminators, we extend Cr-GANs to its conditional version, i.e., Cr-cGANs, by introducing GPM to the original unconditional discriminators D_k of Cr-GANs. The GPM aims to extract effective CI from the mixture. It should be noted that extracting effective CI is essential for SVE. This is because unlike most other generative tasks [21] where we usually have more freedom to generate samples[1], in SVE task, the

[1]For example, for a 'dog' label, we can generate many specific dog samples [26].

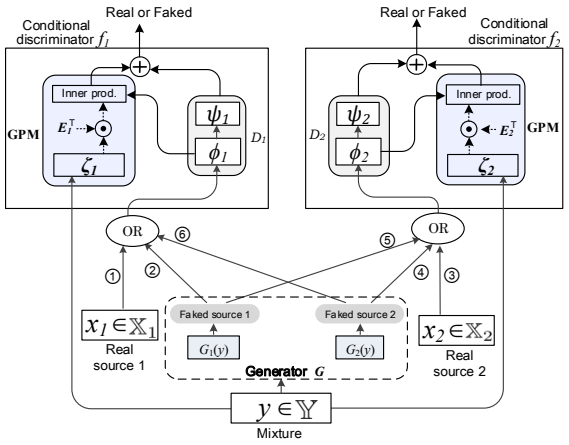

Figure 2: *The proposed Cr-cGANs.*

freedom of generating effective sources in the target distribution with a specific input mixture is quite limited. As a result, the conditional discriminators for SVE need to be highly picky to restrict the generator, especially when the conditional discriminators are trained with unlabelled datasets in a semi-supervised fashion, i.e., the mixtures and sources for training are not paired.

The proposed Cr-cGANs is shown in Fig. 2. In Cr-cGANs, the original unconditional discriminator $D_k(k = 1, 2)$ in Cr-GANs is decomposed into two sub-processes: (i) the intrinsic feature extraction subprocess $\boldsymbol{\phi}_k : \mathbb{X}_k \to \mathbb{R}^{L_1}$ and (ii) the unconditional discriminating subprocess $\psi_k : \mathbb{R}^{L_1} \to \mathbb{R}$, and then D_k can be written as $D_k(z) = \psi_k(\boldsymbol{\phi}_k(z))$, where z is a real or faked source. With this decomposition, the intrinsic features $\boldsymbol{\phi}_k(z)$ can be used for (1) unconditional discriminating, i.e., $\psi_k(\boldsymbol{\phi}_k)$, by modeling the property of each source and (2) conditional discriminating by working cooperatively with effective CI extracted from the mixture y.

In GPM, effective high-level CI (L_2-dimension) is extracted from the mixture $y \in \mathbb{Y}$ using a deep neural network (DNN), $\boldsymbol{\zeta}_k : \mathbb{Y} \to \mathbb{R}^{L_2}$, which has 6 convolutional layers and a dense layer. We compute the inner product of $\boldsymbol{\zeta}_k(y)$ and the intrinsic features $\boldsymbol{\phi}_k(z)$ to incorporate CI into the conditional discriminators of Cr-cGANs.

Using the above GPM, the conditional discriminator of Cr-cGANs, denoted by $f_k(k = 1, 2)$, can be formulated as

$$f_k(z, y) := \boldsymbol{\zeta}_k(y)^{\mathrm{T}} E_k \boldsymbol{\phi}_k(z) + \psi_k(\boldsymbol{\phi}_k(z)), k = 1, 2, \quad (12)$$

where $E_k \in \mathbb{R}^{L_2 \times L_1}$ is the embedding matrix of $\boldsymbol{\zeta}_k(y)$. When $E_k = 0$, $f_k(z, y) = D_k(z)$, i.e., Cr-cGANs reduce to Cr-GANs. When $E_k \neq 0$, f_k considers CI from the mixture.

3.2. Formulation of Cr-cGANs

The proposed Cr-cGANs is constructed by replacing the original unconditional discriminator D_k with the conditional discriminator f_k. According to this implementation, the unsupervised loss L_u in Eq. (3) becomes

$$\tilde{L}_{\mathrm{u}}(\boldsymbol{G}) = -\mathbb{E}_y[f_1(G_1(y), y)] - \mathbb{E}_y[f_2(G_2(y), y)]. \quad (13)$$

The gradient penalty L_k^{grad} in Eq. (5) becomes

$$\tilde{L}_k^{\mathrm{grad}}(x) = \mathbb{E}_y \mathbb{E}_{x|y} \max(||\nabla_x f_k(x, y)||_2 - 1, 0)^2. \quad (14)$$

The conditional adversarial loss in Cr-cGANs is

$$\min_{f_1, f_2} \quad -\tilde{W}_p(f_1, f_2) + \lambda \tilde{P}_{\mathrm{adv}}(f_1, f_2), \quad (15)$$

Algorithm 1 The training algorithm of Cr-cGANs (Cr-GANs).

1: **for** each of the training iterations **do**
2: Fix $\boldsymbol{G}$ and train discriminators f_1 (D_1) and f_2 (D_2) for N_{disc} steps:
adversial : $\min_{f_1,f_2} -\tilde{W}_p + \lambda \tilde{P}_{\mathrm{adv}}$ $(\min_{D_1,D_2} -W_p + \lambda P_{\mathrm{adv}})$,
crossfire : $\min_{f_1,f_2} -\tilde{V}_c + \lambda \tilde{P}_{\mathrm{cr}}$ $(\min_{D_1,D_2} -V_c + \lambda P_{\mathrm{cr}})$.
3: Fix both discriminators and train the generator $\boldsymbol{G}$:
$\min_{\boldsymbol{G}} L_{\mathrm{s}} + \alpha \tilde{L}_{\mathrm{u}} + \beta L_{\mathrm{add}}$ $(\min_{\boldsymbol{G}} L_{\mathrm{s}} + \alpha L_{\mathrm{u}} + \beta L_{\mathrm{add}})$.
4: **end for**

Table 2: *Different models covered by Algorithm 1.*

Models	**Baseline [20]**	**GANs [20]**	**cGANs**	**Cr-GANs**	**Cr-cGANs**
Adversarial	×	√	√	√	√
Crossfire	×	×	×	√	√
GPM	×	×	√	×	√

$$\begin{aligned} \tilde{W}_p(f_1, f_2) &= \mathbb{E}_y \mathbb{E}_{x_1|y}[f_1(x_1, y)] - \mathbb{E}_y[f_1(G_1(y), y)] \\ &+ \mathbb{E}_y \mathbb{E}_{x_2|y}[f_2(x_2, y)] - \mathbb{E}_y[f_2(G_2(y), y)] \quad (16) \\ \tilde{P}_{\mathrm{adv}}(f_1, f_2) &= \tilde{L}_1^{\mathrm{grad}}(\hat{x}_1^{\mathrm{adv}}) + \tilde{L}_2^{\mathrm{grad}}(\hat{x}_2^{\mathrm{adv}}). \quad (17) \end{aligned}$$

Similarly, the conditional crossfire criterion in Cr-cGANs is

$$\min_{D_1, D_2} \quad -\tilde{V}_c(D_1, D_2) + \lambda \tilde{P}_{\mathrm{cr}}(D_1, D_2), \quad (18)$$

$$\begin{aligned} \tilde{V}_c(f_1, f_2) &= \mathbb{E}_y \mathbb{E}_{x_1|y}[f_1(x_1, y)] - \mathbb{E}_y[f_1(G_2(y), y)] \\ &+ \mathbb{E}_y \mathbb{E}_{x_2|y}[f_2(x_2, y)] - \mathbb{E}_y[f_2(G_1(y), y)], \quad (19) \\ \tilde{P}_{\mathrm{cr}}(f_1, f_2) &= \tilde{L}_1^{\mathrm{grad}}(\hat{x}_1^{\mathrm{cr}}) + \tilde{L}_2^{\mathrm{grad}}(\hat{x}_2^{\mathrm{cr}}). \quad (20) \end{aligned}$$

4. Training of Cr-cGANs/Cr-GANs

The training of Cr-cGANs/Cr-GANs is described in Algorithm 1: after each generator updates, we take $N_{\mathrm{disc}}(= 5)$ updating steps for both discriminators, which include a dual-objective optimization made of *adversarial* and *crossfire* steps. All different models covered by Algorithm 1 are summarized in Table 2, where 'Baseline' can be obtained by setting $\alpha = 0$ and $\beta = 0$.

5. Evaluations

For the SVE task, Cr-GANs (Cr-cGANs) have one generator and two discriminators (for the vocal and accompaniment (acc.), respectively). The magnitude and phase of the mixture/source signal were calculated by short-time Fourier transform (STFT) and only the magnitudes were used for training. At inference, the time-domain vocal and acc. sources were obtained by applying inverse STFT to the estimated vocal and acc. magnitudes and the original phase of the mixture, respectively. The separation performance was measured by the BSS-EVAL toolkit [27] with respect to three criteria, source-to-distortion ratio (SDR), source-to-interferences ratio (SIR), and sources-to-artifacts ratio (SAR).

The proposed method was compared with the state-of-the-art GANs based SVE method in [20] under the exactly same conditions. In accordance with [20], all models in Table 2 were evaluated on DSD100 [24] and CCMixter [25] datasets. Table 3 compares the separation performance of all models, where 'Baseline' and 'GANs' comes from [20], 'cGANs' (with GPM), 'Cr-GANs' (with 'crossfire' criterion), and 'Cr-cGANs' (with

Table 3: *Comparisons of the mean performance for different models ($\alpha = \beta = 0.001$) (in dB).*

Dataset	Criteria	**Baseline [20]**		**GANs [20]**		**cGANs**		**Cr-GANs**		**Cr-cGANs (acc)**		**Cr-cGANs (voc)**		**Cr-cGANs (full)**	
		Acc.	Vocal	Acc.	Vocal	Acc.	Vocal	Acc.	Vocal	Acc.	Vocal	Acc.	Vocal	Acc.	Vocal
DSD100	SDR	11.14	3.64	11.19	3.87	11.35	3.99	10.85	3.92	11.32	3.93	**11.39**	3.93	11.26	**4.01**
	SIR	14.23	7.79	14.63	9.32	14.73	9.49	14.55	10.25	**15.12**	9.35	14.88	9.14	14.26	**10.33**
	SAR	14.49	**6.75**	14.24	6.12	14.40	6.13	13.6	5.73	13.99	6.13	14.29	6.26	**14.71**	5.82
CCMixter	SDR	10.87	3.38	11.14	3.69	10.97	3.58	10.66	3.77	11.06	3.68	11.12	3.72	**11.17**	**3.87**
	SIR	15.16	6.70	16.16	8.44	16.14	8.17	15.96	8.89	**16.74**	8.51	16.48	8.39	15.71	**9.02**
	SAR	13.62	**7.97**	13.39	6.86	13.16	7.12	12.88	6.75	13.05	6.83	13.29	7.02	**13.82**	6.96

Table 4: *Comparisons of the mean performance between GANs and Cr-cGANs under different α ($\beta = 0.001$) (in dB).*

Dataset	Criteria	**0.0001**				**0.001**				**0.01**				**0.1**			
		Acc.		Vocal		Acc.		Vocal		Acc.		Vocal		Acc.		Vocal	
		GANs	Cr-cGANs	GANs	Cr-cGANs	GANs	Cr-cGANs	GANs	Cr-cGANs	GANs	Cr-cGANs	GANs	Cr-cGANs	GANs	Cr-cGANs	GANs	Cr-cGANs
DSD100	SDR	11.31	11.33	3.80	4.04	11.19	11.26	3.87	4.01	9.98	11.38	3.08	3.84	9.23	11.23	1.71	4.12
	SIR	15.01	14.80	9.03	10.0	14.63	14.26	9.32	10.33	12.84	15.14	8.25	8.72	11.41	14.84	7.96	9.51
	SAR	14.05	14.33	6.13	5.93	14.24	14.71	6.12	5.82	13.58	14.13	5.58	6.45	13.85	14.06	3.95	6.30
CCMixter	SDR	10.89	10.87	3.42	3.45	11.14	11.17	3.69	3.87	9.97	10.71	2.78	3.25	9.88	10.84	2.68	3.68
	SIR	16.31	16.31	7.97	8.39	16.16	15.71	8.44	9.02	14.16	15.95	7.17	7.77	13.40	16.28	8.17	7.89
	SAR	13.00	13.02	6.93	6.79	13.39	13.82	6.86	6.96	12.76	13.02	6.33	6.83	13.32	12.99	5.16	7.39

both GPM and 'crossfire' criterion) are the proposed models. We trained the 'Cr-cGANs' model in three ways (see the last three columns): 'Cr-cGANs (acc)' used only the acc. discriminator, 'Cr-cGANs (voc)' used only the vocal discriminator, and 'Cr-cGANs (full)' used both the acc. and vocal discriminators. The hyper-parameters were set as $\alpha = \beta = 0.001$ [20].

We can observe from Table 3 that

(i) cGANs vs. GANs: when applying GPM to GANs, (i.e., the cGANs model), the SDR, SIR, and SAR on DSD100 were improved for both vocal and acc., while the results on CCMixter were degraded;

(ii) Cr-GANs vs. GANs: when adding the 'crossfire' criterion to GANs (i.e., the Cr-GANs model), the SDR and SIR of vocal were improved for both datasets;

(iii) Cr-cGANs (full) vs. GANs: when both GPM and 'crossfire' criterion were applied (i.e., the Cr-cGANs model), we obtained an overall improvement for SDR and SIR on vocal for both two datasets; and

(iv) Cr-cGANs (full) vs. cGANs and Cr-GANs: the Cr-cGANs performed much better than only applying the GPM or 'crossfire' criterion, indicating that the GPM and 'crossfire' criterion were beneficial to each other.

In addition, the Cr-cGANs (full) model was better than Cr-cGANs (acc) and Cr-cGANs (voc) models on most metrics for the CCMixter dataset, which suggested that two 'crossfire' discriminators could work cooperatively to eliminate the mutual inferences from the interfering source.

We also compared the performance of Cr-cGANs and GANs using different α from 0.0001 to 0.1, where β was fixed at 0.001. According to Table 4, the Cr-cGANs was better than GANs for both datasets, especially for $\alpha = 0.01, 0.1$.

Finally, we compared the statistics between GANs and Cr-cGANs (full). Except for mean (Mean) and standard deviation (SD), the median (Med) results with its median absolute deviation (MAD) were calculated as they were more robust against outliers [28]. It can be seen from Table 5 that the Cr-cGANs was superior to GANs in terms of SDR, e.g., 0.21 dB improvement in Med for DSD100 and 0.46 dB improvement for CCMixter. For SIR, we obtained 0.59 dB improvement in Med and 1.01 dB improvement in Mean for DSD100, and similar results for CCMixter. For SAR, although the mean of Cr-cGANs for DSD100 is 0.3 dB lower than GANs, the Med of Cr-cGANs still slightly increased. For CCMixter, the SAR of Cr-cGANs were 0.35 dB in Med and 0.1 dB in Mean better than GANs.

Table 5: *Statistics of singing voice extracting (in dB).*

Vocal		SDR		SIR		SAR	
		GANs	Cr-cGANs	GANs	Cr-cGANs	GANs	Cr-cGANs
DSD100	Med	4.61	**4.82**	10.21	**10.8**	6.38	**6.44**
	MAD	1.23	**0.95**	1.95	**1.68**	0.98	**0.77**
	Mean	3.87	**4.01**	9.32	**10.33**	**6.12**	5.82
	SD	2.77	**2.61**	**3.55**	3.58	2.03	**1.96**
CCMixter	Med	4.11	**4.57**	8.91	**9.13**	7.23	**7.58**
	MAD	2.0	**1.95**	**2.58**	3.28	2.2	**1.72**
	Mean	3.69	**3.87**	8.44	**9.02**	6.86	**6.96**
	SD	4.06	4.06	**5.4**	5.66	**2.6**	2.69

6. Conclusions

We introduced a novel GANs/cGANs framework, i.e., Cr-GANs/Cr-cGANs, for the SVE task. The separation performance of Cr-GANs is enhanced by jointly training with the traditional adversarial criterion and the proposed crossfire criterion, where the crossfire criterion allows the discriminators to work cooperatively with the generator for eliminating the mutual interferences from the interfering source. To effectively use conditional information (CI), we updated the Cr-GANs to its conditional version, i.e., Cr-cGANs, by introducing the GPM to the unconditional discriminators. Experimental results have shown the effectiveness of Cr-GANs/Cr-cGANs. These frameworks can be potentially extended to the scenario of three or more sources and other audio source separation applications.

7. Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 61902280), the Natural Science Foundation of Tianjin (No. 19JCYBJC15600), the Tianjin Science and Technology Project (No. 20YDTPJC00870), the Tianjin Major Project for Civil-Military Integration of Science and Technology (No. 18ZXJMTG00260). It was also supported by a Grant-in-Aid for Scientific Research (B) (No. 17H01761), I-O DATA foundation, and the Fund for the Promotion of Joint International Research (Fostering Joint International Research (B)) (20KK0233).

8. References

[1] E. J. Humphrey, S. Reddy, P. Seetharaman, A. Kumar, R. M. Bittner, A. M. Demetriou, S. Gulati, A. Jansson, T. Jehan, B. Lehner, A. Krupse, and L. Yang, "An introduction to signal processing for singing-voice analysis: High notes in the effort to automate the understanding of vocals in music," *IEEE Signal Processing Magazine*, vol.36, no.1, pp. 82–94, 2019.

[2] Z. Rafii, A. Liutkus, F. Ster, S. I. Mimilakis, D. FitzGerald, and B. Pardo, "An overview of lead and accompaniment separation in music," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 8, pp. 1307–1335, 2018.

[3] Y. Ikemiya, K. Itoyama, and K. Yoshii, "Singing voice separation and vocal F0 estimation based on mutual combination of robust principal component analysis and subharmonic summation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 11, pp. 2084–2095, 2016.

[4] B. Zhu, W. Li, R. Li, and X. Xue, "Multi-stage non-negative matrix factorization for monaural singing voice separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 10, pp. 2096–2107, 2013.

[5] E. M. Grais and H. Erdogan, "Source separation using regularized NMF with MMSE estimates under GMM priors with online learning for the uncertainties," *Digital Signal Processing*, vol. 29, pp. 20–34, 2014.

[6] G. Hu and D. L. Wang, "A tandem algorithm for pitch estimation and voiced speech segregation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing.*, vol. 18, no. 8, pp. 2067–2079, 2010.

[7] A. S. Bregman, *Auditory Scene Analysis. Cambridge*. MA, USA: MIT Press, 1990.

[8] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MA, USA: MIT Press, 2016.

[9] A. Narayanan and D. Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 7092–7096.

[10] Y. Wang and D. Wang, "Towards scaling up classification-based speech separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 7, pp. 1381–1390, 2013.

[11] Y. Wang, A. Narayanan, and D. Wang, "On training targets for supervised speech separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 12, pp. 1849–1858, 2014.

[12] S. Uhlich, F. Giron, and Y. Mitsufuji, "Deep neural network based instrument extraction from music," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 2135–2139.

[13] P. S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Deep learning for monaural speech separation," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 1562–1566.

[14] S. I. Mimilakis, K. Drossos, G. Schuller, and T. Virtanen, "A recurrent encoder-decoder approach with skip-filtering connections for monaural singing voice separation," in *Proceedings of the IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2017, pp. 1–6.

[15] S. I. Mimilakis, K. Drossos, J. F. Santos, G. Schuller, T. Virtanen, and Y. Bengio, "Monaural singing voice separation with skip-filtering connections and recurrent onference of time-frequency mask," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 721–725.

[16] C. Subakan and P. Smaragdis, "Generative Adversarial Source Separation," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 26–30.

[17] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proceedings of the Advances in Neural Information Processing Systems*, 2014, pp. 2672–2680.

[18] H. Choi, J. Lee, and K. Lee, "Singing voice separation using generative adversarial networks," in *NIPS Machine Learning for Audio (ML4Audio) Workshop*, 2017.

[19] Z. Fan,Y. Lai, and J. Jang, "SVSGAN: singing voice separation via generative adversarial network," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 726–730.

[20] D. Stoller, S. Ewert, and S. Dixon, "Adversarial semi-supervised audio source separation applied to singing voice extraction," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 2391–2395.

[21] T. Miyato and M. Koyama, "cGANs with projection discriminator," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[22] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, "Improved training of Wasserstein GANs," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems*, 2017, pp. 5769–5779.

[23] M. Arjovsky and L. Bottou, "Towards principled methods for training generative adversarial networks," *arXiv preprint 1409.0876*, 2017.

[24] A. Liutkus, F. R. Stöter, Z. Rafii, D. Kitamura, B. Rivet, N. Ito, N. Ono, and J. Fontecave, "The 2016 signal separation evaluation campaign," in *Proceedings of the International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA)*, 2017, pp. 323–332.

[25] A. Liutkus, D. Fitzgerald, and Z. Rafii, "Scalable audio separation with light kernel additive modelling," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 76–80.

[26] A. Brock, J. Donahue, and K. Simonyan, "Large scale GAN training for high fidelity natural image synthesis," in *Proceedings of the 7th International Conference on Learning Representations (ICLR)*, 2019.

[27] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–1469, 2006.

[28] D. Stoller, S. Ewert, and S. Dixon, "Wave-U-Net: A Multi-Scale Neural Network for End-to-End Audio Source Separation," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, 2018, pp. 334–340.