FOLEY SOUND SYNTHESIS BASED ON GAN USING CONTRASTIVE LEARNING WITHOUT LABEL INFORMATION

Technical Report

Hae Chun Chung, Yuna Lee, Jae Hoon Jung

KT Corporation, Republic of Korea

ABSTRACT

Sound effects used in radio or movies, such as foley sound, have been difficult to create without the help of experts. Furthermore, in the field of audio synthesis, the field of speech has been actively progressed, but there has been no research on audio sounds that can be obtained in real life. In this technical report, We present our submission system for DCASE2023 Task7: Foley-sound synthesis. We participate in track B, which forbids the usage of external resources. We propose a framework that employ the loss function of ContraGAN and C-SupConGAN based on structure of Self-Attention GAN (SAGAN). Our final system achieves outperforming the baseline performance by a large margin.

Index Terms— Foley sound synthesis, Generative Adversarial Network, Contrastive Learning

1. INTRODUCTION

Foley sound is a term used to describe sound effects that are created to convey and enhance the sounds produced by events, especially in a narrative such as radio or film [1]. There has been a number of research on generating desired sounds [2, 3, 4]. However, they were mainly focused on voice synthesis based on singing, text-to-speech (TTS), and music generation rather than acoustic domain like sound effects or background noises. Few researches has shown attention to detecting background noises or sound effects in previous DCASE challenges such as task6b, but they were limited to audio-tagging fields or audio captioning, which describes more specific details in text [5, 6]. In the following DCASE 2023 challenges, task 7: Foley sound synthesis were created to break new ground of the audio synthesis in creating user-desired sound suitable for user-defined environments [7]. The following task 7 consists of two subtask A and B. The use of outside resources is where the two differ from one another. In this technical report, we participated in subtask B, which do not use any external sources.

For the DCASE 2023 challenge, we proposed two-stage system based on Generative Adversarial Network (GAN), which is a powerful tool for generation task in a variety of domains. The first stage of the system aims to map a sound category input, such as 'dog bark', to a Mel spectrogram. Our system can be diverged into two types in terms of processing this phase. First, the following framework adopts adversarial loss function and conditional contrastive loss (2C loss) of ContraGAN [8] which applies data-to-data and data-to-class relationship in the discriminator. Second, we use adversarial loss function and conditional supervised contrast loss function (C-SupCon loss), which is derived from C-SupConGAN [9]. We use audio features extracted from pre-trained audio encoder network for C-SupCon loss [9]. The audio encoder was pre-trained with supervised contrastive learning [10].

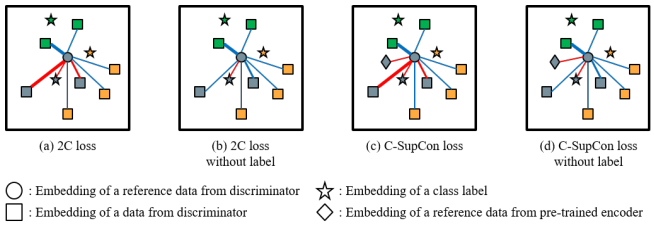

Figure 1: Our overall idea of optimizing data-to-data distances, data-to-class distance and data-to-source distance. The color of each shape represents a class. The color of line implies the push-and-pull between the embeddings. Red line represents pulling each embeddings while blue line represents pushing each other. The thickness of the line expresses the strength of the pushing and pulling force. The thicker the line, the stronger the pull or push.

In the second stage, we use fixed vocoder network of HiFi-GAN[11] suggested by the challenge to produce more robust results rather than proposing novel network. This first framework shows FAD of 5.060, and the other strategy could achieve FAD of 4.833. These performances outperform the baseline overall FAD score.

The rest of the report is organized as follows. In the Section 2, we present C-SupConGAN, which is the key framework in our system, and its variants in detail. Section 3 outline the experiments for performance evaluation, and we analyze our proposed system and its performance in the section 4. Finally, we summarize our system and discuss future work in the section 5.

2. METHODS

We used aforementioned 2-stage system to obtain high performance of FAD score in this task. Our overall framework is described in Figure 2. We denote the first stage as 'category-to-sound' section and the second as 'Mel spectrogram-to-sound' section for straightforward explanation. Since we suggest two distinct systems, the differences between two systems will be discussed further along with the loss function during the first stage.

2.1. Category-to-Mel spectrogram

In the first stage, we applied C-SupConGAN for category-to-Mel spectrogram synthesis in the first stage of category-to-sound generation. C-SupConGAN's main model structure is Self-Attention

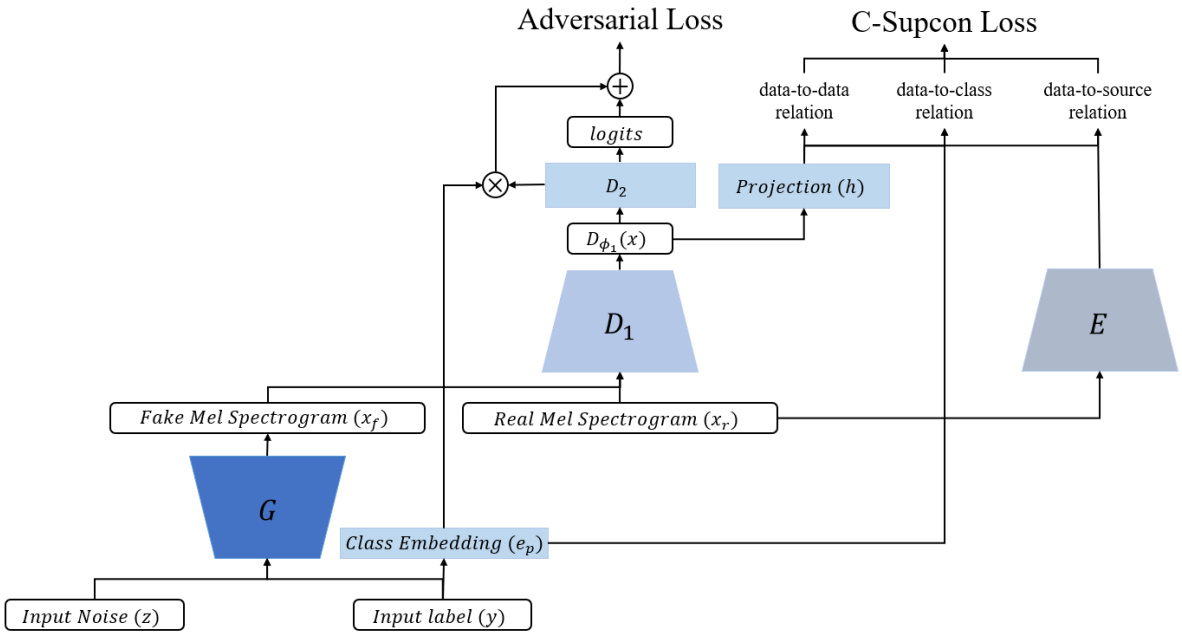

Figure 2: Our overall framework.

GAN (SAGAN), which adds C-SupCon loss, an improved form of 2C loss, to adversarial loss.

Adversarial Loss GAN is composed of generator and discriminator. Generator G intend to deceive the discriminator D with synthetic Mel spectrogram generated from the given label information. On the other hand, the discriminator D must establish the validity of the generated Mel spectrogram and the real Mel spectrogram using label information. Thus, G takes noise z_i with label information of class i, c_i, while D takes real Mel spectrogram x_i or fake Mel spectrogram $G(z_i, c_i)$ based on the same label information c_i. We use the hinge loss function as the adversarial loss function, and each objective functions for D and G are shown in the equation below.

$$l_D = -\min\left(0, -1 + D\left(x_i, c_i\right)\right) - \min\left(0, -1 - D\left(G\left(z_i, c_i\right), c_i\right)\right) \tag{1}$$

$$l_G = -D\left(G\left(z_i, c_i\right), c_i\right)$$

Conditional Contrastive Loss (2C loss) ContraGAN, which is one of the aspired framework of our systems, incorporated 2C loss to stabilize GAN training. 2C loss is a supervised method that minimizes data-to-data distance and data-to-class distance belong to same class and maximizes data-to-data distance belong to different class via extracted features of data embedding from the discriminator. As shown in Figure 2, we divided the discriminator D into two separate networks: D_1 and D_2. We extract real or fake data embedding d_i from the $D_1(\cdot)$ and the projection head $h(\cdot)$ while class embedding $e(c_i)$ is extracted by embedding function $e(\cdot)$. Through cosine similarity equations, these features are mapped to unit hypersphere.

Although we applied 2C loss, the fact that the number of classes is small leads to the unexpected situation. We discovered that the adversarial loss of the discriminator D falls too quickly when we implement the 2C loss function as it is. This occurrence leads to the poor GAN training, eventually to mode collapse problem [12] that produces similar outputs within the class. To resolve this catastrophic event, we exclude label from the original 2C loss function. Therefore, the model is optimized in a way that data-to-data distances belong to both same class and different class are maximized while data-to-class distances of same class are minimized. This modification induce stable training of GAN, securing the variance of generated class-wise outputs. The following data-to-data distance $d2d_{i,j}$ and data-to-class $d2c_{i,i}$ can be denoted as the equation 3.

$$d2d_{i,j} = \exp\left(d_i \cdot d_j / \tau_d\right), \; d2c_{i,i} = \exp\left(d_i \cdot e(c_i) / \tau_c\right) \tag{2}$$

With aforementioned notation, the modified 2C loss function is defined as follows:

$$l_{2C}\left(d_i, c_i\right) = -\log\left(\frac{d2c_{i,i}}{d2c_{i,i} + \sum_{k=1}^{N} 1_{i \neq k} \cdot d2d_{i,j}}\right) \tag{3}$$

The $\cdot$ symbol denotes the inner (dot) product, and N is batch size. The hyperparameter τ is applied to control the pushing and pulling forces; the larger τ, the weaker the force, and the smaller τ, the stronger the force. As C-SupConGAN differentiates the temperature for data-to-data distance τ_d and data-to-class distance τ_c for boost performance, we also set each temperature hyperparameter differently. By default, we set $\tau_d = 0.1, \tau_c = 1.0$, which is used in C-SupConGAN. The comparison between the training guidance of the 2C loss function with and without label information is schematically depicted in (a) and (b) of Figure 1.

C-SupCon loss By adjusting 2C loss and adversarial loss in training GAN, ContraGAN strengthen GAN's robustness to training collapse problem. Nevertheless, ContraGAN still holds the instability

in training process such as tackling with training collapse after certain steps, ultimately end in rapid drop in performance. To mitigate this restraint, we extend our GAN framework to C-SupConGAN. C-SupconGAN is distinctive in the fact that there is an additional pretrained encoder network $E(\cdot)$. Through reference data embeddings extracted from $E(\cdot)$, C-SupConGAN adds a data-to-source relation to the standard 2C loss function. This aided GAN's feature learning, reduced the instability of training process, enabling long-term training, and ultimately improved the performance. Therefore, we utilize the C-SupCon loss, an advanced version of the 2C loss, to improve performance further. We also removed the label information in C-SupCon loss.

$$d2s_{i,i} = \exp\left(d_i \cdot f(d_i)/\tau_c\right) \tag{4}$$

In the same way, the modified C-SupCon loss can be described as follows:

$$l_{C-SupCon}\left(d_i, c_i\right) = -\log\left(\frac{d2s_{i,i} + d2c_{i,i}}{d2s_{i,i} + d2c_{i,i} + \sum_{k=1}^{N} 1_{i\neq k} \cdot d2d_{i,k}}\right) \tag{5}$$

The cases in which label information is used and excluded in the C-SupCon loss function can be visually confirmed in (c) and (d) of Figure 1.

For implementation of encoder network $E(\cdot)$, we used ResNet18 [13] as the encoder network, and it was pretrained with Supervised Contrastive Learning (SupCon) [10] loss function. For audio augmentation, we used fade in/out and time masking during pretraining process. After pretraining process is completed, we proceed classification finetuning and classification evaluation. Since additional dataset such as evaluation dataset was not open to public, we could only evaluate the performance of classification on training set. The classification accuracy achieved 100%, which may appear as overfitting, but we can infer that the pretrained encoder network $E(\cdot)$ is capable of extracting high quality audio embeddings from the training set. Thus, we denote the relationship between mel spectrogrm of real audio x_r and extracted feature embedding vector $E(x_r)$ as data-to-source distance of the C-SupCon loss.

Therefore, our total system is optimized through two types of loss function, which is the combination of adversarial loss and 2C loss function and the combination of adversarial loss and C-SupCon loss function. 2C loss or C-SupCon loss is expressed as l_C. In this way, total loss function $\mathcal{L}$ can be described:

$$\mathcal{L}_D = \frac{1}{N}\sum_{k=1}^{N} l_D + \frac{1}{N}\sum_{k=1}^{N} l_C,\ \mathcal{L}_G = \frac{1}{N}\sum_{k=1}^{N} l_G + \frac{1}{N}\sum_{k=1}^{N} l_C \tag{6}$$

$$\mathcal{L} = \mathcal{L}_D + \mathcal{L}_G \tag{7}$$

2.2. Mel spectrogram-to-sound

After the training on the first phase, Generator network G have the ability to generate Mel spctrogram from class category. During the second stage, pretrained vocoder network transforms the generated Mel spectrogram into a time-domain digital audio signal. Instead of proposing novel vocoder network, we apply the pretrained vocoder network.

3. EXPERIMENT

We devise our experiments for two purposes. First, we conduct experiments to show that our two proposed techniques surpass the baseline system. Second, we design experiments to verify the impact of label information on our general framework and to on ablation study. We submit 4 different systems for the challenge. Since we have two different baselines, we pin $\tau_d = 0.1$ in each baseline, then $\tau_c = 0.1$ and $\tau_c = 1.0$. All four systems use the same implementation details as follows.

3.1. Experiment metrics

We use Frechet Audio Distance (FAD) [14]. FAD is a standard metrics for music enhancement, and very useful in that it is a reference-free evaluation metric. FAD can be employed even in the absence of a ground truth reference audio because it is calculated from collections of hidden representations of created and real samples. The FAD score can be computed by multivariate Gaussians between the generated data set and the actual audio data set, which can be referred as the reference embeddings.

3.2. Implementation Details

We use the log mel-band energies of input audio as audio feature. We set the frame length to 1024, and hop size as 256. All the models we train are devised to generate 80×344 mel spectrogram. By default, the learning rate for generator is 0.0001 and the learning rate for the discriminator is 0.0001. Initially, We used the same learning rate value applied in C-SupConGAN. However, the small amount of dataset lead to the circumstance of discriminator D learning too quickly. Thus, we set borth learning rates to 0.0001. For all models, we use Adam optimizer with $\beta_1 = 0.5$ and $\beta_2 = 0.999$ for training. We build 2-layer projection layer $h(\cdot)$ which embeds the output of the portion of discriminator network D_1 to 128-dimension. During training, we freeze the weight of pretrained encoder network $E(\cdot)$.

3.3. Dataset

The DCASE 2023 task 7 development set contains 4,850 labeled sound fragments, which can be classified into 7 categories: dog bark, footstep, gunshot, keyboard, moving motor vehicle, rain, and sneeze/cough. Each sound was fitted to a length of 4 seconds, and zero-padded or segmented if necessary. All audio was transferred to mono 16-bit 22,050 Hz sampling rate [7]. As we are participating in subtask B, we do not use any external sources.

4. RESULTS

4.1. Ablation Study

Table 1 shows the comparison result of the existence of label information in baseline frameworks affect the performance. Table 1

	w. label	**w/o** label
2C loss	12.667	5.060
C-SupCon loss	12.552	4.833

Table 1: The comparison of FAD score on two baseline.

depict the performance when we apply the label information in the loss function during the training process on our frameworks. From

	DogBark	Footstep	GunShot	Keyboard	MovingMotorVehicle	Rain	Sneeze/Cough	Average FAD
Jung_S0	2.899	4.149	4.821	3.411	14.929	3.848	1.449	5.072
Jung_S1	2.829	3.807	3.634	4.222	15.673	3.534	1.717	5.060
Jung_S2	2.559	3.414	5.985	3.468	12.591	4.219	2.390	4.947
Jung_S3	2.749	3.765	4.913	2.867	14.364	3.709	1.466	4.833

Table 2: The comparison of FAD score on the submitted systems. S0, S1,S2,S3 are four systems we submitted to the challenge.

the table, we can claim that excluding label information in training process can achieve noticeable performances. We speculate this consequences as follows. When label information exists, the model optimizes in a way that data-to-data distance and data-to-class distance belongs to the same class is minimized while data-to-data distance belongs to different class is maximized. Unlike the task to which C-SupConGAN or 2C loss was applied, the number of class and the quantity of training dataset of the task7 are scarce. This attribute leads to situation which data belongs to the same class are densly clustered refraining the diversity of individual data within the following class. Therefore, the loss of discriminator *D* drop rapidly, eventually results poor training of GAN. This calamity leads us to exclude label information. Under the absence of label information, the model is trained in a way that data-to-data distance of all classes, same and different, are maximized and still maintain the trait of the class as the data-to-class distance remains. This cause data to secure the characteristic of class but enlarge the diversity between individual data at the same time. The following event makes training difficult and proceed to better training. Therefore, higher quality of data is generated. Accordingly, we argue that excluding label information during training stage lead to superior performance in this DCASE 2023 challenge task7-B.

4.2. Performance Comparison

In Table 2, we compare the four systems for average FAD and class-wise FAD, respectively. The results in the table demonstrate the C-SupCon loss performs better than 2C loss function. We claim that as C-SupCon loss adds data-to-source relation to 2C loss using pre-trained features, which supports the feature learning of GAN and further stabilizes the training process.

Jung_S0 and Jung_S2 are set $\tau_d = 0.1, \tau_c = 0.1$, Jung_S1 and Jung_S3 are set $\tau_d = 0.1, \tau_c = 1.0$. As can be seen in Table 2, increasing τ_c from 0.1 to 1.0 showed greater performance. What τ_c increases is to reduce the strength of the data-to-class relation. Compared to other classes, the 'GunShot' class dropped more than 1 when $\tau_c = 1.0$ than when $\tau_c = 0.1$. We postulate this phenomenon is based on the characteristic of the 'GunShot' class. Due to the various traits such as each type of gun or the number of rounds shot exhibits, the data contains high degree of acoustic diversity within same class. Decreasing the strength of the data-to-class relationship increases the diversity of data features belonging to the same class while retaining the characteristics of the class to which the data features belong. This will lead to improved performance. Accordingly, we can infer that increasing τ_c can enlarge the variances of each data feature within class, which leads to secure the diversity of synthesized audio samples within class.

Table 3 refers to performance comparison between baseline method with our proposed methods: 2C loss and C-SupCon loss. Our two techniques outperform baseline methods in every way. In particular, in 'DogBark' and 'Rain' classses, our baseline frameworks performed 4 to 5 times better than the existing baseline. We speculate that this remarkable performance is due to the proposed frameworks' ability to enhance variance of data features within the class while keeping distinct characteristic of class using our proposed loss function. In Table 3, we can see that improvement of FAD performance of class 'Moving Motor Vehicle' is rather low. We infer this outcome is based on insufficient variance of audio data within the class. This trait induce generation of similar data in the class regardless of the methods. To sum up, our proposed frameworks achieve the average FAD score of 5.060 and 4.833, which is the half of the baseline.

Class	Baseline	**Ours**	
		2C	**C-SupCon**
DogBark	13.411	2.829	2.749
Footstep	8.109	3.807	3.765
GunShot	7.951	3.634	4.913
Keyboard	5.230	4.222	2.867
MovingMotorVehicle	16.108	15.673	14.364
Rain	13.337	3.534	3.709
Sneeze/Cough	3.770	1.717	1.466
Average FAD	9.702	5.060	4.833

Table 3: The FAD score on each class index.

5. DISCUSSION

In this report, we arranged two types of framework for the DCASE 2023 challenge task7-B. Our methods are based on ContraGAN and C-SupConGAN, and aim to secure the particular class features while securing the distinctiveness of individual data within class by data-to-data relations, data-to-class relations, and data-to-source relations. The frameworks we propose show the performance of achieving a FAD score of 4.833 and 5.060, which outperform the existing baseline by a large margin. Still, our methods fall to generate more diverse audio samples from classes with low data diversity, such as 'Moving Motor Vehicle' class provided in the development set. We intend to supplement this part through future work.

6. REFERENCES

[1] http://dcase.community/challenge2023/.

[2] Y. Hono, K. Hashimoto, K. Oura, Y. Nankaku, and K. Tokuda, "Singing voice synthesis based on generative adversarial networks," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6955–6959.

[3] N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimberg, A. Oord, S. Dieleman, and K. Kavukcuoglu, "Efficient neural audio synthesis," in *International Conference on Machine Learning*. PMLR, 2018, pp. 2410–2419.

[4] J. Engel, C. Resnick, A. Roberts, S. Dieleman, M. Norouzi, D. Eck, and K. Simonyan, "Neural audio synthesis of musical notes with wavenet autoencoders," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1068–1077.

[5] Q. Kong, Y. Xu, W. Wang, and M. D. Plumbley, "Sound event detection of weakly labelled data with cnn-transformer and automatic threshold optimization," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2450–2460, 2020.

[6] X. Mei, X. Liu, J. Sun, M. D. Plumbley, and W. Wang, "Diverse audio captioning via adversarial training," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2022.

[7] K. Choi, J. Im, L. Heller, B. McFee, K. Imoto, Y. Okamoto, M. Lagrange, and S. Takamichi, "Foley sound synthesis at the dcase 2023 challenge," *In arXiv e-prints: 2304.12521*, 2023.

[8] M. Kang and J. Park, "Contragan: Contrastive learning for conditional image generation," *Advances in Neural Information Processing Systems*, vol. 33, pp. 21 357–21 369, 2020.

[9] H. Chung and J.-K. Kim, "C-supcongan: Using contrastive learning and trained data features for audio-to-image generation," in *Proceedings of the 2022 5th Artificial Intelligence and Cloud Computing Conference*, 2022, pp. 135–142.

[10] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," *Advances in neural information processing systems*, vol. 33, pp. 18 661–18 673, 2020.

[11] J. Kong, J. Kim, and J. Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," *Advances in Neural Information Processing Systems*, vol. 33, pp. 17 022–17 033, 2020.

[12] H. Thanh-Tung and T. Tran, "Catastrophic forgetting and mode collapse in gans," in *2020 international joint conference on neural networks (ijcnn)*. IEEE, 2020, pp. 1–10.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*. Springer, 2016, pp. 630–645.

[14] K. Kilgour, M. Zuluaga, D. Roblek, and M. Sharifi, "Fréchet audio distance: A reference-free metric for evaluating music enhancement algorithms." in *INTERSPEECH*, 2019, pp. 2350–2354.