SOUND SCENE SYNTHESIS BASED ON GAN USING CONTRASTIVE LEARNING AND EFFECTIVE TIME-FREQUENCY SWAP CROSS ATTENTION MECHANISM

Technical Report

Hae Chun Chung

KT Corporation
Acoustic Processing Project, AI Tech Lab
Seoul, Republic of Korea
hc.chung@kt.com

Jae Hoon Jung

KT Corporation
Acoustic Processing Project, AI Tech Lab
Seoul, Republic of Korea
hoony.jung@kt.com

ABSTRACT

This technical report outlines the efforts of KT Corporation's Acoustic Processing Project for addressing sound scene synthesis, DCASE 2024 Challenge Task 7. The task's objective is to develop a generative system capable of synthesizing environmental sounds from text descriptions. Our system is designed in three stages to achieve this goal: embedding the text description, generating a mel spectrogram conditioned on the text embedding, and converting the mel spectrogram into an audio waveform. Our main focus lies on training the model for the second stage. We employed a generative adversarial network (GAN) and meticulously designed the training process and architecture. We utilized various contrastive losses and introduced the single-double-triple attention mechanism to accurately capture text descriptions and train high-quality features. To mitigate the rise in GPU memory consumption caused by the expanded attention mechanism, we designed a novel time-frequency swap cross-attention mechanism. Our system achieved FAD score more than 30% lower than the DCASE baseline, demonstrating significant performance improvements in text-to-audio generation.

Index Terms— text-to-audio generation, generative adversarial networks, contrastive learning, attention mechanism

1. INTRODUCTION

The world is filled with an array of sounds, each originating from diverse sources, each with its own unique characteristics. In a sound scene, these various sounds are intertwined in a complex manner. These sound scenes are often laid out as background audio in various recorded media to further maximize the auditory effect. Accurately capturing the desired sound scene can be resource-intensive, demanding either on-location recording or meticulous studio creation. Sound scene generation by AI system could drastically reduce costs. However, research in this field is still nascent, and the performance of existing models remains suboptimal. The DCASE 2024 Challenge Task 7: Sound Scene Synthesis [1] targets this issue. This task focuses on developing the system that can create sound scenes based on textual descriptions. This technical report details the model and approach we used to address this task.

We designed a three-stage system for sound scene synthesis. In the first stage, text embedding is extracted from textual description using text encoder. The text encoder pre-trained with contrastive language-audio pre-training (CLAP) [2] was used. In the second stage, we employed generative adversarial networks (GAN) [3], to map text embedding to mel spectrogram. In the third stage, HiFi-GAN [4], a pre-trained vocoder, was used to convert the generated mel spectrogram into audio waveform. We focused the GAN model training on the second stage. The GAN training process and architecture were meticulously designed. In addition to adversarial loss, multiple contrastive losses were introduced for training high-quality features and closely matching the text description. Furthermore, we propose a new single-double-triple attention structure that combines audio self-attention, audio-to-sentence cross-attention, and audio-to-word cross-attention. To mitigate the increase in GPU memory consumption caused by the expanded attention mechanism, we designed a novel time-frequency swap cross attention (TF-SCA) that significantly reduces GPU memory consumption while optimizing the characteristics of audio feature data. Our system achieved fréhet audio distance (FAD) [5] of 41.2, which is improvement of over 30% improvement compared to the baseline model's FAD of 61.28 provided in the challenge.

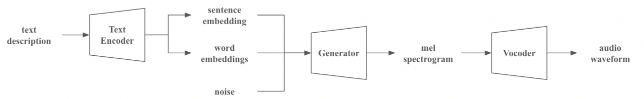

Figure 1: Overall system.

2. METHODS

2.1. Overview

We designed a three-stage system. In the first stage, a given text description y is fed into the text encoder to extract word embeddings w and a sentence embedding s. In the second stage, the generator creates a mel-spectrogram $\hat{m}$ using w, s, and noise z. Finally, in the third stage, $\hat{m}$ is input into the vocoder to convert to the final audio waveform $\hat{x}$.

We used a text encoder pre-trained with contrastive language-audio pre-training (CLAP) and a pre-trained HiFiGAN as the vocoder. Our focus was on designing and training the generative adversarial networks (GAN) to generate a mel-spectrogram from the given text information w and s. Therefore, the architectural design of the GAN and its training process will be described in detail.

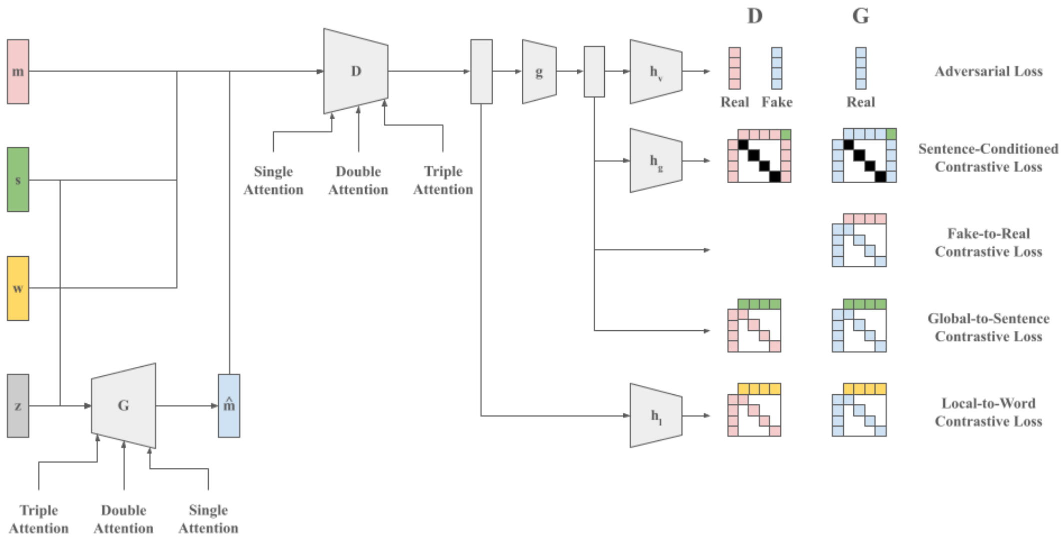

Figure 2: Overview of GAN training.

2.2. The training of the GAN

The GAN is composed of generator and discriminator. The generator aims to create data that the discriminator cannot distinguish from real data, while the discriminator's goal is to differentiate between real data and the data generated by the generator. These two models train adversarially. As training progresses, the generator creates increasingly realistic data, making it progressively harder for the discriminator to distinguish between real and generated data.

In addition to the adversarial loss used for training the GAN to generate data, we introduced various contrastive losses to stabilize training and generate text-conditioned data. To utilize features for contrastive losses, we divided the discriminator into three parts : the discriminator encoder D , which map mel-spectrogram to 3-dimensional local feature, the global sum pooling g, which compresses the 3-dimensional local feature into 1-dimensional global feature, and the linear projection head h_v, which maps the 1-dimensional feature into a single value. An overview of our GAN's training process is shown in Figure 2.

Adversarial Loss Let (x_i, y_i) be the i-th audio-text pair data of randomly sampled minibatch N. x_i is converted to a mel-spectrogram m_i, y_i is fed into the text encoder to extract word embeddings w_i and sentence embedding s_i. The generator G takes noise z_i along with w_i and s_i to create a mel-spectrogram $\hat{m}_i = G(z_i, w_i, s_i)$. The discriminator D determines the authenticity of either the real data pair $d_i = (m_i, w_i, s_i)\, or fake data pair \hat{d}_i = (\hat{m}_i, w_i, s_i)$. We use the hinge loss function [6] as the adversarial loss function, and each objective functions for D and G are shown in the equation below.

$$\begin{aligned}&\mathcal{L}_D = -\min(0, -1 + h_v(g(D(d_i)))) - \min(0, -1 - h_v(g(D(\hat{d}_i))))\\&\mathcal{L}_G = -h_v(g(D(\hat{d}_i)))\end{aligned} \tag{1}$$

Sentence-Conditioned Contrastive Loss To stabilize GAN training and generate conditional data, several conditional contrastive losses were employed [7, 8, 9, 10, 11]. We applied a conditional contrastive loss function using global feature $f_i^g = g(D(d_i))$ and sentence embedding. Multi-layer perceptrons were used as a projection head h_g to map the global feature to a new unit hypersphere. The equation is as follows.

$$\begin{aligned}&\mathcal{L}_{SC}\left(f_i^g, w_i; \tau\right) = \\&-\log \frac{\exp\left(h_g(f_i^g) \cdot w_i/\tau\right)}{\exp\left(h_g(f_i^g) \cdot w_i/\tau\right) + \sum_{k=1}^{N} 1_{i \neq k} \cdot \exp\left(h_g(f_i^g) \cdot h_g(f_k^g)/\tau\right)}\end{aligned} \tag{2}$$

where $\cdot$ denotes the dot product, and τ is the temperature parameter. This approach encourages the generation of text-conditioned data by enhancing the similarity with the corresponding sentence embedding while ensuring data diversity by reducing the similarity between all data in the mini-batch.

Fake-to-Real Contrastive Loss The primary method for the generator to determine the similarity between its generated data and real data is the adversarial loss, derived from the single numerical result of the discriminator. To support this, we introduce a feature contrastive loss that directly compares the generated data with the real data in the feature space. The equation is as follows.

$$\mathcal{L}_{F2R}\left(\hat{f}_i^g, f_i^g; \tau\right) = -\log \frac{\exp(\hat{f}_i^g \cdot f_i^g/\tau)}{\sum_{k=1}^{N} \exp(\hat{f}_i^g \cdot f_k^g/\tau)} \tag{3}$$

This encourages the features $\hat{f}_i^g = g(D(\hat{d}_i))$ of the generated data to be similar to those of the reference real data, while simultaneously ensuring they are not similar to the features of other real data. As a result, the generator creates realistic and diverse data that closely resembles real data while maintaining unique characteristics.

Global-to-Sentence Contrastive Loss The generator must produce data that is both realistic and aligned with the text description. To achieve this, sentence features are used as conditions in both the discriminator and the generator, but this alone is insufficient. Therefore, we introduced a contrastive loss between data features and sentence features in the feature space. This approach strengthens the discriminator's ability to evaluate data according to the sentence and reinforces the generator's capacity to create data that better matches the sentence description.

$$\mathcal{L}_{G2S}\left(f_i^g, s_i; \tau\right) = -\log \frac{\exp(f_i^g \cdot s_i/\tau)}{\sum_{k=1}^{N}\exp(f_i^g \cdot s_k/\tau)} \tag{4}$$

This loss maximizes the similarity between the global features of the data and the reference sentence features while minimizing the similarity to other sentence features. This encourages the generation of data that is well-suited to the text description.

Local-to-Word Contrastive Loss The global-to-sentence contrastive loss effectively incorporates the overall impression of the text description into the data, but it may lack the precision to convey finer details. Each word in the text description carries significant meaning. Therefore, specific adjustments are required for local parts of the generated data according to the individual words. To achieve this, we employ a contrastive loss with attention mechanisms that learn connections between local regions of the data and specific words in the text without requiring fine-grained annotations that align each word with its corresponding local region [12, 7]. This method ensures that the generated data accurately involves both the overall context and the detailed nuances of the text description. For k^{th} local features $f_k^l = D(d_k)$ and word features w_k in minibatch, the soft attention $\alpha_{k,i,j}$ for the i^{th} word feature $w_{k,i}$ to the j^{th} local feature $f_{k,j}^l$ is calculated as follows.

$$\alpha_{k,i,j} = \frac{\exp(\gamma_1(w_{k,i} \cdot f_{k,j}^l))}{\sum_{r=1}^{R}\exp(\gamma_1(w_{k,i} \cdot f_{k,r}^l))} \tag{5}$$

where R is the total number of local regions in the local feature and γ_1 is a smoothing hyper-parameter to reduce the entropy of the soft attention. The aligned local feature for the i^{th} word is defined as $c_{k,i} = \sum_{r=1}^{R}\exp(\alpha_{k,i,j} \cdot f_{k,r}^l)$. The score function between all the local regions in local feature f_k^l and all words in word feature w_k can the be defined as:

$$\mathcal{S}(f_k^l, w_k) = -\log\left(\sum_{t=1}^{T}\exp(\gamma_2(w_{k,t} \cdot c_{k,t}))\right)^{\frac{1}{\gamma_2}} \tag{6}$$

where T is the total number of words in the word feature w_k, and γ_2 is a hyper-parameter that determines the weight of the most aligned word-region pair. When $\gamma_2 \to \infty$, $\mathcal{S}(f_k^l, w_k)$ approximates to $\max_{t=1}^{T}\mathcal{S}(f_{k,t}^l, w_{k,t})$. Finally, the contrastive loss between the words and local regions in local feature f_k^l and its aligned word feature w_k can be defined as:

$$\mathcal{L}_{L2W}\left(f_k^l, s_k; \gamma_1, \gamma_2, \gamma_3\right) = -\log \frac{\exp(\gamma_3 \mathcal{S}(f_k^l, w_k))}{\sum_{h=1}^{N}\exp(\gamma_3 \mathcal{S}(f_k^l, w_h))} \tag{7}$$

where γ_3 is a smoothing hyper-parameter. To reduce loss, the word-region score meticulously calculates the correlation between each region within the data's local features and each word in the text's word features.

2.3. The architecture of the GAN

We delves deeply into both the training methodology and the architectural design of the GAN model. The architecture of the GAN is based on the OC-SupConGAN [11] with robust generation capability. The class feature was replaced with the sentence feature and the model size is increased to enhance its capacity. To better capture the details of the text description, the existing self-attention was extended to single-double-triple attention. However, increasing the attention mechanism leads to an increase in GPU memory consumption. To alleviate this, we designed a time-frequency swap cross attraction (TF-SCA) that is optimized for the characteristics of audio features and mitigates the GPU memory consumption.

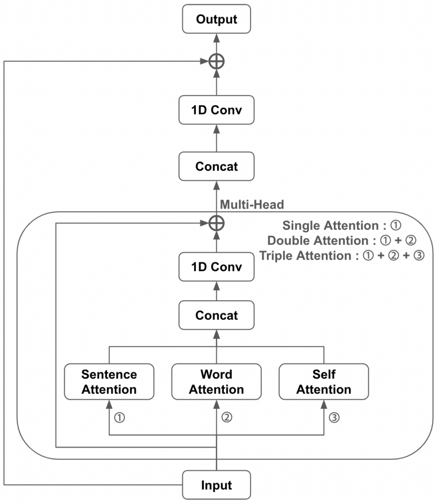

Figure 3: Triple Attention.

Single-Double-Triple Attention To enhance feature quality, we incorporate self-attention and cross-attention mechanisms. Self-attention assigns weights to features within the audio data, prioritizing those with greater significance. Cross-attention focuses on capturing important information from one modality by attending to the other. This allows for effective fusion of information across two different modality.

We used three types of attention: audio self-attention (ASA), audio-to-sentence cross-attention (A2S-CA), audio-to-word cross-attention (A2W-CA). These attention mechanisms can be utilized in varying combinations. We used them in three ways: single attention (ASA), double attention (A2S-CA, A2W-CA), and triple attention (A2S-CA, A2W-CA, and ASA), which are depicted in Figure 3. This was applied depending on the feature size within the model. For the generator, the order progresses from triple attention to double attention to single attention. Conversely, the discriminator utilizes them in the opposite order, starting with single attention and progressing to triple attention.

Time-Frequency Swap Cross Attention Self-attention offers many advantages in feature training, but it has the drawback of significantly increasing GPU memory consumption due to the size of the attention weight matrix. Audio feature has three dimensions: time (T), frequency (F), and channel (C). Using traditional self-attention on audio feature ($T \times F \times C$), size of an attention weight matrix is $(T \times F) \times (T \times F) = T^2F^2$, which would be very large. One approach to alleviate this issue involves applying separate attention mechanisms to the time and frequency dimensions (time-attention and frequency-attention). This reduces the size of the attention weight matrix to $(T \times T) + (F \times F) = T^2 + F^2$. However, audio feature is inherently a combination of time and fre-

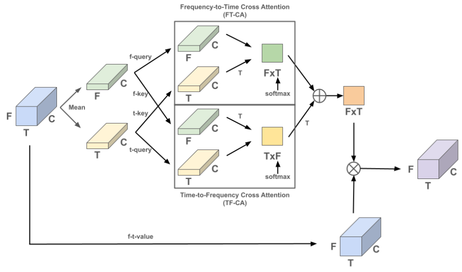

Figure 4: Self Time-Frequency Swap Cross Attention.

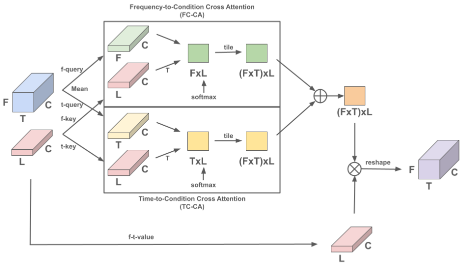

Figure 5: Multi Time-Frequency Swap Cross Attention.

quency, rather than treating them separately as independent dimensions.

To address this limitation, we propose a novel time-frequency swap cross attention (TF-SCA) mechanism. This approach capitalizes on the inherent interplay between time and frequency in audio features. TF-SCA achieves this by performing cross-attention, where the time and frequency dimensions are swapped: time-to-frequency cross attention and frequency-to-time cross attention, as illustrated in Figures 4.

$$T^2F^2 \geq T^2 + F^2 \geq 2TF \;\; (T > 1, F > 1) \tag{8}$$

This not only better captures the combined nature of audio feature but also significantly reduces the size of the attention weight matrix to $(F \times T) + (T \times F) = 2TF$. As a result, GPU memory consumption for the attention mechanism is significantly lower than with the two aforementioned methods. Furthermore, the TF-SCA mechanism can be extended for multi-modal cross-attention, as illustrated in Figures 5.

3. SETTING

3.1. Training Data

For model training, we used AudioCaps dataset (AC) [13], and data belonging to the AudioSet in the WavCaps datasets (WC) [14]. All data samples consist of a consistent duration of 10 seconds and a sampling rate of 32 kHz. To use the pre-trained HiFi-GAN using 48 kHz data, all data were upsampled to 48 kHz and converted them into 256-bin mel spectrograms with a frame length of 2048 and a hop size of 480. As a result, 256×1024 mel spectrograms are derived.

3.2. Model

The text encoder for embedding the text and the vocoder for converting the mel spectrogram into waveform used pre-trained models, and all parameters were frozen. CLAP was used as the text encoder and HiFi-GAN was used as the vocoder. The learning rate for generator is 0.0001 and the learning rate for the discriminator is 0.0004. The number of updates in the discriminator per number of updates in the generator is 2. For all models, we use Adam optimizer [15] with $\beta_1 = 0.5$ and $\beta_2 = 0.999$ for training. τ was all set to 0.1, and γ_1, γ_2, and γ_3 were set to 5, 5, and 50, respectively.

3.3. Inference

The required format for the audio output in the challenge is 32 kHz with a duration of 4 seconds, while the format of the audio for training is 48 kHz with a duration of 10 seconds. To bridge this disparity, we followed a three-step process to derive the final output.

Firstly, we generated 10 audio waveforms for a text prompt, each with a duration of 10 seconds and a sample rate of 48 kHz, using generative system. Subsequently, the pre-trained CLAP model was used to calculate the similarity between the audio outputs and the text prompt, and one audio output with the highest similarity was selected. Second, we randomly chopped the audio output, resulting in 10 audio results with a duration of 4 seconds. For these 10 audios, the audio result with the highest similarity to the text prompt is selected. The final audio result was downsampled from 48 kHz to 32 kHz.

3.4. Test Data

We tested 60 text prompts, the development set provided in DCASE2024 Challenge Task7 [1]. The text prompt describes the foreground and background.

3.5. Metric

Fréhet Audio Distance (FAD) using PANNs CNN14 Wavegram-Logmel (pans-wavegram-logmel) [16] embedding selected by the challenge coordinators side was used as a metric [17]. The lower the FAD value, the better the performance.

4. RESULTS

The generator creates different outputs each time for the same text input due to random noise, resulting in variations in performance measurements. Therefore , to evaluate the average performance of the model, we averaged the results from 10 runs.

	Baseline	Our system
FAD	61.276	42.075

Table 1: The comparison of FAD score.

The baseline provided in the challenge achieved a FAD score of 61.28, whereas our system achieved a FAD score of 42.075. This represents a remarkable performance improvement of over 30% compared to the baseline. Moreover, upon listening to the audio generated by our system, they delicately capture the content of the text prompt and produce high-quality sound. This demonstrates that our system is meticulously designed and serves as a high-performance model for text-to-audio generation.

5. REFERENCES

[1] http://dcase.community/challenge2024/task-sound-scene-synthesis.

[2] Y. Wu, K. Chen, T. Zhang, Y. Hui, T. Berg-Kirkpatrick, and S. Dubnov, "Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[3] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[4] 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.

[5] 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.

[6] J. H. Lim and J. C. Ye, "Geometric gan," *arXiv preprint arXiv:1705.02894*, 2017.

[7] H. Zhang, J. Y. Koh, J. Baldridge, H. Lee, and Y. Yang, "Cross-modal contrastive learning for text-to-image generation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 833–842.

[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] J. Jeong and J. Shin, "Training gans with stronger augmentations via contrastive discriminator," *arXiv preprint arXiv:2103.09742*, 2021.

[10] 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.

[11] H. Chung, Y. Lee, and J. Jung, "Foley sound synthesis based on generative adversarial networks using oneself-conditioned contrastive learning."

[12] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He, "Attngan: Fine-grained text to image generation with attentional generative adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1316–1324.

[13] C. D. Kim, B. Kim, H. Lee, and G. Kim, "Audiocaps: Generating captions for audios in the wild," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 119–132.

[14] X. Mei, C. Meng, H. Liu, Q. Kong, T. Ko, C. Zhao, M. D. Plumbley, Y. Zou, and W. Wang, "Wavcaps: A chatgpt-assisted weakly-labelled audio captioning dataset for audio-language multimodal research," *arXiv preprint arXiv:2303.17395*, 2023.

[15] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[16] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "Panns: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.

[17] M. Tailleur, J. Lee, M. Lagrange, K. Choi, L. M. Heller, K. Imoto, and Y. Okamoto, "Correlation of fr\'echet audio distance with human perception of environmental audio is embedding dependant," *arXiv preprint arXiv:2403.17508*, 2024.