FROM NOISE TO SOUND: AUDIO SYNTHESIS VIA DIFFUSION MODELS

Technical Report

Haojie Zhang[1,2], *Kun Qian*[1,2*], *Lin Shen*[1,2], *Lujundong Li*[1,2], *Kele Xu*[3*], *Bin Hu* [1,2*],

[1] Key Laboratory of Brain Health Intelligent Evaluation and Intervention,
Ministry of Education (Beijing Institute of Technology), P. R. China
[2] School of Medical Technology, Beijing Institute of Technology, P. R. China
zhj@bit.edu.cn, qian@bit.edu.cn, richers0129@163.com, lljd@bit.edu.cn, bh@bit.edu.cn
[3] National University of Defense Technology, xukelele@163.com

ABSTRACT

In this technical report, we describe our submission system for DCASE2023 Task 7: Foley Sound Synthesis (Track B). A Sound Pixelate Diffuse model is proposed to realize foley sound synthesis. The model includes data format conversion and synthesising audio through the diffusion model. The Synthesised audio are evaluated on DCASE2023 Task 7 Eval FAD evaluation set and the best FAD score of all categories is 8.429.

***Index Terms*—** DCASE, Foley Sound Synthesis, Generative Model, Diffusion Model

1. INTRODUCTION

Intelligent audio generation refers to the process of automatically generating audio content through computer systems using artificial intelligence technology and algorithms. It combines technologies such as audio processing, machine learning, and deep neural networks to simulate and replicate the ability of human audio creation. Audio generation has great application prospects in fields such as media background music and intelligent music intervention [1]. Foley sound [2] is the art of creating sound effects for media using props and methods to simulate everyday sounds. It is a crucial aspect of post production and requires specialized approaches for generating a large number of similar yet distinct sounds[3].

The mainstream methods for sound generation can be categorized into two types [4, 5, 6]: Generative Adversarial Network (GAN), and Diffusion Model. GANs produce high-quality output but can be prone to mode collapse [7, 8]. As an alternative, Diffusion Model employs a more stabler training algorithm called inverse diffusion, which facilitates easier training and avoids mode collapse. Additionally, the Diffusion Model does not require post-training updates, making the generation process more efficient and eliminating the need to retain the generator model. So it can be used for Foley sound synthesis due to their stable training and ability to generate diverse and novel samples [9].

In this work, we propose a diffusion-model-based model named Sound Pixelate Diffuse (SPD) for sound synthesis. Furthermore, we employed wavelet filtering to reduce noise in the audio signal, which may effectively decompose the signal into its various frequency components, facilitating targeted noise reduction in each component.

2. METHODS

2.1. Denoising Diffusion Implicit Model

Denoising Diffusion Probabilistic Model (DDPM) is a probabilistic modeling approach used to generate high-quality images or videos. It achieves this by gradually diffusing and denoising noise images [10]. Specifically, DDPM uses an increasing diffusion coefficient to control the diffusion speed of noise images and applies a noise-reconstruction network to denoise the image at each step. DDPM models the diffusion process using an inverse differential equation. The equation 1 is the sampling process of DDPM.

$$X_{t-1} = \frac{1}{\sqrt{X_t}}(X_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha_t}}}\epsilon_\theta(X_t, t)) + \sigma_t Z \tag{1}$$

DDPM's noise addition is based on the Markov chain process, which means that the denoising process must also be based on this process, leading to a large number of steps. Compared to DDPM, Denoising Diffusion Implicit Model (DDIM) proposed Non-Markovian forward processes and derived a faster sampling process based on this assumption [11]. To achieve the best possible sound quality, we ultimately chose to use DDIM for Foley sound synthesis. Unlike DDPM, the equation 2 reduces the number of sampling steps by decreasing the dependence on the posterior distribution.

$$X_{t-1} = \sqrt{\alpha_{t-1}}(\frac{X_t - \sqrt{1-\alpha_{t-1}}\epsilon_\theta(X_t)}{\sqrt{\alpha_t}}) + D_t + \sigma_t\epsilon_t \tag{2}$$

Here D_t stands for direction pointing to X_t.

$$D_t = \sqrt{1-\alpha_{t-1}-\sigma_t^2} \cdot \epsilon_\theta(X_t) \tag{3}$$

2.2. Sound Pixelate Diffuse Model

Training models directly on audio waveforms can be computationally expensive and time-consuming due to the high dimensionality and complex nature of raw audio date. So we attempted to convert

*This work was partially supported by the Ministry of Science and Technology of the People's Republic of China with the STI2030-Major Projects (No. 2021ZD0201900), the National Natural Science Foundation of China (No. 62227807 and 62272044), and the Teli Young Fellow Program from the Beijing Institute of Technology, China. (*Corresponding authors: K. Qian, K. Xu and B. Hu.*)

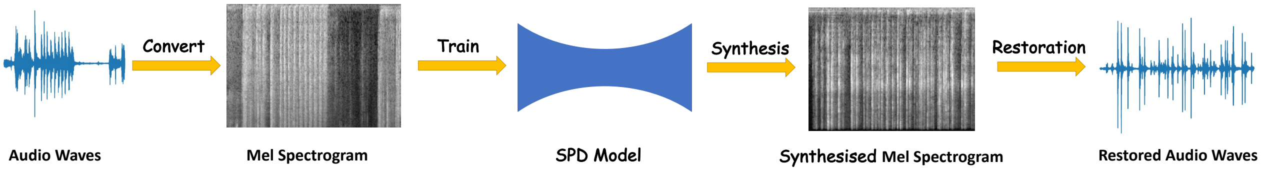


Figure 1: Framework of SPD model for audio generation.

the audio file into a Mel Spectrogram image and then used this image to convert it into an audio file. After such restoration, we found that the audio obtained did not differ significantly in terms of the listening experience and effect [12].

Figure 1 shows the framework of SPD model. In the SPD model, the Mel Spectrogram image format is used as the input, which can bring some advantages [13]. Using the Mel Spectrogram as the input can greatly reduce data volume and processing time, and improve the efficiency of training and processing [14]. Furthermore, the Mel filter can convert the original signal into a more linear signal with the human ear's perception of sound frequency, which helps reduce the impact of noise and improve the auditory perception of audio processing. We utilize the well-established DDIM model and we introduce a distinct modification to the U-Net architecture. This modification involves transforming the previously square-shaped receptive fields into rectangular ones, which is specifically designed to better align with the requirements of our task. This adaptation aims to enhance the model's ability to generate outputs that effectively meet the task's demands.

3. EXPERIMENT

3.1. Data Preprocessing

The raw audio signal has a sampling frequency of 22,050 Hz, and we extract input features from it using Short Time Fourier Transformation (STFT) with a window size of 1024 and 25% overlap. We also apply an 80-band Log Mel filter bank to process the original signal [15, 16].

3.2. SPD Model Settings

The diffusion model framework is trained for a total of 300 epochs. AdamW optimizer with a weight decay of 1e-6 and a starting learning rate of 1e-4 is utilized, along with betas of (0.95, 0.999) as the momentum coefficients for the Adam optimization algorithm to balance the effects of the first and second moments of the gradients. The batch size is set to 8. It allows for good results to be achieved after running for approximately 12 hours on an NVIDIA A40 GPU. This approach results in a significant improvement in training speed compared to the official baseline.

The input images have a size of (256, 386). So we design a network that is suitable for rectangular shapes. The input feature map is down-sampled using multiple DownBlock2D modules to reduce the resolution while increasing the number of channels to extract more feature information. An AttentionBlock layer is introduced in the fifth layer to incorporate an attention mechanism that better captures the importance of different features in the input, thereby improving the accuracy and generalization performance of the model. Subsequently, the low-resolution feature map is up-sampled to the original resolution using multiple UpBlock2D layers while utilizing more high-level features.

Table 1 lists the architecture and detailed parameters of the SPD model.

Table 1: The network architecture and parameters of SPD model.

Block	kernel stride
DownBlock2D_1	conv, 3x3, (128,128,192)
DownBlock2D_2	conv, 3x3, (128,64,96)
DownBlock2D_3	conv, 3x3, (256,32,48)
DownBlock2D_4	conv, 3x3, (256,16,24)
DownBlock2D_5+AttentionBlock	conv, 3x3, (512,8,12)
DownBlock2D_6	conv, 3x3, (512,4,6)
MidBlock2D	conv, 3x3, (512,4,6)
UpBlock2D_1	conv, 3x3, (512,4,6)
UpBlock2D_2+AttentionBlock	conv, 3x3, (512,8,12)
UpBlock2D_3	conv, 3x3, (256,16,24)
UpBlock2D_4	conv, 3x3, (256,32,48)
UpBlock2D_5	conv, 3x3, (128,64,96)
UpBlock2D_6	conv, 3x3, (128,128,192)

3.3. Sound Synthesis

The SPD model only requires a small number of iterations to generate audio results that meet the required standards. After 50 iterations, the model produced satisfactory diffusion results. Due to the alteration of data volume during the pre-processing stage, our model demonstrated high efficiency during audio generation, taking only 2 seconds to generate a single audio by using a single commercial grade NVIDIA RTX A5000 GPU.

After obtaining the generated audio data using SPD, we attempted to reduce noise in the audio through wavelet filtering. Wavelet transforms possess excellent time-frequency properties, enabling the decomposition of signals into different frequency components [17]. As such, we employed wavelet filtering as a means of reducing noise in our audio signal. By utilizing wavelet transforms, we were able to effectively decompose the signal into its various frequency components, facilitating targeted noise reduction in each component. We employed the sym8 wavelet filter with a fixed thresholding method (sqtwolog) and a wavelet transform level set to 5, to filter the output of our model.

From the perspective of the Mel Spectrogram, although wavelet denoising yields cleaner and purer audio in terms of generation. The filtering effect is shown in figure 3. In terms of experimental results, we observed that wavelet filtering had a mixed effect on the FAD values: depending on the signal characteristics, it could either increase or decrease the FAD values. Consequently, we adopted a

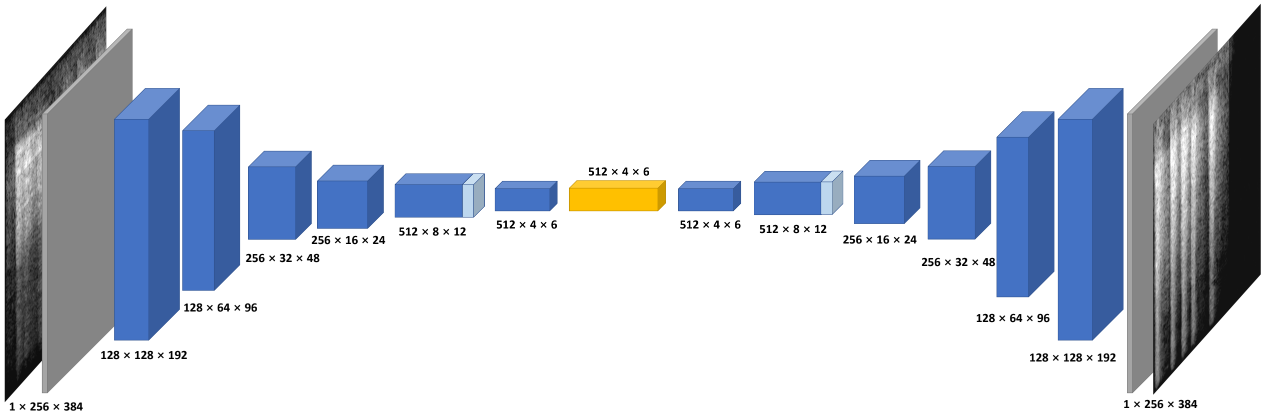
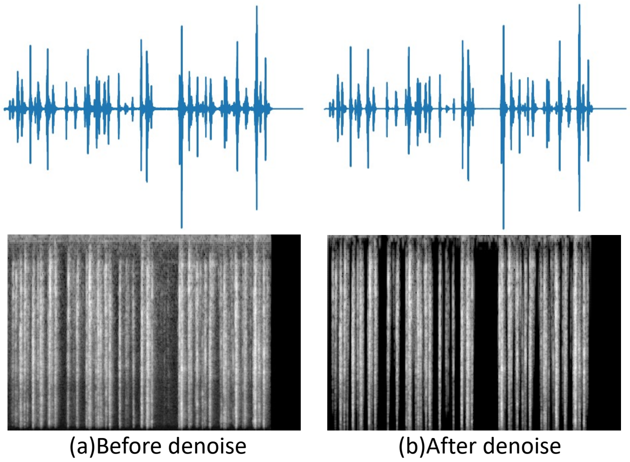


Figure 2: Schematic diagram of SPD model structure.


(a)Before denoise

(b)After denoise

Figure 3: Wavelet domain denoise.

selective approach in our submission, applying noise reduction to some of the submissions while leaving others unfiltered.

4. SUBMISSIONS AND RESULTS

The best final results are reported in Table 2. Below we describe our submissions in detail:

Submission 1: This submission uses the results of the 250 epochs of training. No denoising was performed on the generated audio.

Submission 2: This submission uses the results of the 300 epochs of training. No denoising was performed on the generated audio.

Submission 3: This submission uses the results of the 500 epochs of training. No denoising was performed on the generated audio.

Submission 4: This submission uses the results of the 300 epochs of training. Using Wavelet domain denoise on the generated audio.

Table 2: The best final results.

Category	FAD
DogBark	8.866
Footstep	5.478
GunShot	9.333
Keyboard	4.936
MovingMotorVehicle	14.488
Rain	5.647
Sneeze/Cough	10.259
ALL	8.429

5. CONCLUSION

This technical report provides a brief overview of the progress in Track 7 of the DCASE 2023 Challenge. We utilized a diffusion model to construct the entire model and converted the audio into images to improve training speed and generation quality while maintaining satisfactory results. Additionally, we applied a wavelet denoising technique to the generated audio to achieve optimal auditory effects. By combining these optimization techniques, our model achieves faster training speed while ensuring generation quality. This allows us to train the model more quickly and generate higher-quality audio. This is particularly important for real-time applications and large-scale generation tasks as it improves efficiency and achieves significant improvements in quality.

6. REFERENCES

[1] K. Qian, B. W. Schuller, X. Guan, and B. Hu, “Intelligent music intervention for mental disorders: Insights and perspectives,” *IEEE Transactions on Computational Social Systems*, vol. 10, no. 1, pp. 2–9, 2023.

[2] 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.

[3] C. Gan, D. Huang, P. Chen, J. B. Tenenbaum, and A. Torralba, “Foley music: Learning to generate music from videos,” in *Computer Vision–ECCV 2020: 16th European Conference,*

Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16. Springer, 2020, pp. 758–775.

[4] B. Dai and D. Wipf, "Diagnosing and enhancing vae models," *arXiv preprint arXiv:1903.05789*, 2019.

[5] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An overview," *IEEE signal processing magazine*, vol. 35, no. 1, pp. 53–65, 2018.

[6] P. P. Ebner and A. Eltelt, "Audio inpainting with generative adversarial network," *arXiv preprint arXiv:2003.07704*, 2020.

[7] R. Durall, A. Chatzimichailidis, P. Labus, and J. Keuper, "Combating mode collapse in gan training: An empirical analysis using hessian eigenvalues," *arXiv preprint arXiv:2012.09673*, 2020.

[8] Z. Zhang, M. Li, and J. Yu, "On the convergence and mode collapse of gan," in *SIGGRAPH Asia 2018 Technical Briefs*, 2018, pp. 1–4.

[9] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.

[10] J. Choi, S. Kim, Y. Jeong, Y. Gwon, and S. Yoon, "Ilvr: Conditioning method for denoising diffusion probabilistic models," *arXiv preprint arXiv:2108.02938*, 2021.

[11] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020.

[12] A. Mesaros, T. Heittola, and T. Virtanen, "Acoustic scene classification: an overview of dcase 2017 challenge entries," in *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*. IEEE, 2018, pp. 411–415.

[13] E. Moliner, J. Lehtinen, and V. Välimäki, "Solving audio inverse problems with a diffusion model," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[14] N. Chen, Y. Zhang, H. Zen, R. J. Weiss, M. Norouzi, and W. Chan, "Wavegrad: Estimating gradients for waveform generation," *arXiv preprint arXiv:2009.00713*, 2020.

[15] Y. Hwang, H. Cho, H. Yang, D.-O. Won, I. Oh, and S.-W. Lee, "Mel-spectrogram augmentation for sequence to sequence voice conversion," *arXiv preprint arXiv:2001.01401*, 2020.

[16] 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.

[17] B. Kozłowski, "Time series denoising with wavelet transform," *Journal of Telecommunications and Information Technology*, no. 3, pp. 91–95, 2005.