FALL-E: A FOLEY SOUND SYNTHESIS MODEL AND STRATEGIES

Minsung Kang[*], *Sangshin Oh*[*], *Hyeongi Moon, Kyungyun Lee, Ben Sangbae Chon*

Gaudio Lab, Inc., Seoul, South Korea. bc@gaudiolab.com

ABSTRACT

This paper introduces FALL-E, a foley synthesis system and its training/inference strategies. The FALL-E model employs a cascaded approach comprising low-resolution spectrogram generation, spectrogram super-resolution, and a vocoder. We trained every sound-related model from scratch using our extensive datasets, and utilized a pre-trained language model. We conditioned the model with dataset-specific texts, enabling it to learn sound quality and recording environment based on text input. Moreover, we leveraged external language models to improve text descriptions of our datasets and performed prompt engineering for quality, coherence, and diversity. FALL-E was evaluated by an objective measure as well as listening tests in the DCASE 2023 challenge Task 7. The submission achieved the second place on average, while achieving the best score for diversity, second place for audio quality, and third place for class fitness.

Index Terms— Sound synthesis, foley, generative audio

1. INTRODUCTION

Generative AI has seen significant progress in recent years, particularly in the domains of images and text. However, the progress in sound generation has been comparatively slower.

In the field of sound generation, numerous impressive works have been introduced including text-to-sound models such as AudioGen [1] and AudioLDM [2]. In addition, several works can be used as modules of the whole system such as Hifi-GAN [3], SoundStream, EnCodec [4, 5], latent diffusion [6], and spectrogram super-resolution [7].

Furthermore, in text-input and text-conditioned generation, models such as T5 [8], GPT [9, 10], text prompt engineering [11, 12], and diffusion with conditioned generative models [1, 2, 13, 14] have been introduced. As the behavior of large deep learning models is somewhat difficult to analyze, these works enable us as users to steer the model using carefully selected text inputs.

In this context, we present a novel approach to foley synthesis that utilizes a cascade system composed of low-resolution spectrogram generation, a super-resolution module, and a vocoder. Our system represents our submission to the DCASE 2023 Task 7 - Foley Synthesis Challenge (Track A) [15]. While we report objective measures with respect to the official evaluation set, our ultimate goal is to develop sound generation models that extend beyond the challenge's scope.

In Section 2, we introduce our model architecture, FALL-E, detailing the function of each module and how they work in tandem. In Section 3, we provide an in-depth analysis of our evaluation results, showcasing the effectiveness of our approach in various settings. Lastly, in Section 4, we summarize our contributions and highlight future directions for our work.

2. FALL-E

2.1. Architecture

The cascade system, which involves generating low-resolution images or features and subsequently obtaining higher-resolution results, has been extensively utilized in generation models[14, 16, 17]. We adopt this approach to generate foley sound. Our proposed system, FALL-E, consists of three separately-trained models: diffusion-based low-resolution spectrogram generation model and upsampling model, and a GAN-based mel-spectrogram inversion network.

Text Encoder of FALL-E is a pre-trained Flan-T5, an instruction finetuned-variant of a T5 model which shows better performance for various applications [18]. The class category is mapped to predefined text prompts from the prompt corpus. Then Flan-T5 converts the text prompts into a sequence of text embedding, which is input to the Low-resolution Spectrogram Generator.

Low-resolution Spectrogram Generator is based on Glide, a diffusion generative model for text-to-image generation [14]. This module produces a low-resolution spectrogram. Specifically, it generates a 32 × 128 feature map for a 128-bin, 512-frame mel-spectrogram. The module employs a U-Net shaped architecture with 5 residual blocks in both the encoder and decoder. In the encoder, each block comprises 2 convolution layers and an additional upsampling layer with the number of convolution channels in each block increasing linearly from 192. The decoder is a mirrored version of the encoder.

Spectrogram Upsampler is another diffusion-based

[*] Equal contribution.

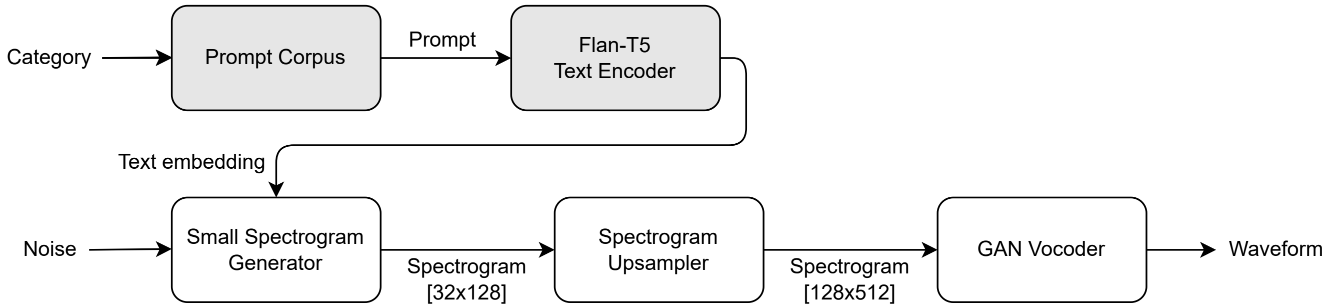

Figure 1: The overall system. Shaded blocks indicate the rule-based or pretrained models.

Module name	Num. of Parameters
Text Encoder	110 M
Low-res. Spec. Generator	318 M
Spectrogram Upsampler	89 M
Mel Inversion Network	125 M
Total	642 M

Table 1: The number of parameters in each module.

generative model that synthesizes mel-spectrograms from a given low-resolution spectrogram. The overall architecture of this model is a U-Net that is similar to Low-resolution Spectrogram Generator but with a different number of blocks and channels. Its encoder and decoder consists of 4 blocks and the number of convolution channel in the first block is 128. Unlike Low-resolution Spectrogram Generator, it isn't conditioned on text; it is only conditioned by the low-resolution mel-spectrogram feature.

Mel Inversion Network converts the generated mel-spectrograms into waveforms. Based on HiFi-GAN [19] and BigVGAN [20], we add FiLM [21] layers as a residual connection. The additional layer helps the model to preserve signal characteristics of the conditioned spectrogram and improves the phase reconstruction quality. We open-sourced this mel inversion network, GOMIN.[1]

The whole system has 642M parameters in total. Its details are described in Table 1.

2.2. Datasets

Training datasets include various sources across private and public audio datasets, including AudioSet [22], CLOTHO [23], FreeToUseSounds.[2], Sonniss,[3] WeSoundEffects,[4] and ODEON.[5] To prevent data imbalances or the potential risks of model misbehavior, samples with speech or musical contents are filtered out based on their metadata. After the filtering, we used 3,815 hours of audio signals for training.

Name	AQ	Dataset Size		Modality		
		Dura.	**N. Files**	**Lb**	**Cp**	**Vd**
Public dataset						
AudioSet	*noisy*	5420 *h*	1,951,460	✓		✓
Clotho	*noisy*	37.0 *h*	5,929		✓	
Free To Use Sounds	*noisy*	175.7 *h*	6,370		✓	
Sonnis Game Effects	*clean*	84.6 *h*	5,049		△	
WeSoundEffects	*clean*	12.0 *h*	488		△	
Odeon Sound Effects	*clean*	19.5 *h*	4,420		△	
Private dataset						
Private dataset	*clean*	3829 *h*	371,116		△	

Table 2: A list of audio datasets. AQ: audio quality, Dura.: duration, N. Files: number of files. Modality columns refer to the existence of labels, captions, and videos, respectively. *Clean* recording: Audio is recorded in well-treated environments and mastered for professional content production. *Noisy*: dataset contains environmental noises or interference signals. △: Textual information included, not necessarily captions. This table is partially from [1] and [24]

2.3. Prompting Strategy

Text conditioning can be optimized or engineered to improve the model behavior. One of our focuses was to control the recording condition/environment of the generated signals so that the model can learn from crowd-sourced, noisy datasets (low recording SNR) as well, while being able to produce high-quality audio. Among the datasets we used, AudioSet, Clotho, and Free To Use Soounds were ”*noisy*” dataset. We append a special token that indicates *noisy dataset* to the text input during training. For the other datasets, we append *clean dataset* token. The impact of this additional token will be discussed in Section 3. We also clean the text label (i.e., text normalization) by dropping some stop words and numbers.

Our model is designed to process natural language text. When we directly use the sound class name as input, we

[1] https://github.com/ryeoat3/gomin
[2] https://www.freetousesounds.com/all-in-one-bundle/
[3] https://sonniss.com/gameaudiogdc
[4] https://wesoundeffects.com/we-sound-effects-bundle-2020
[5] https://www.paramountmotion.com/odeon-sound-effects

Sound class	**WAS ↑**	**Qual. ↑**	**Fit. ↑**	**Div. ↑**	**FAD ↓**
Dog bark	7.984	7.612	8.223	8.250	11.456
Footstep	6.865	6.455	7.082	7.250	5.959
Gun shot	7.255	6.814	7.573	7.500	3.021
Keyboard	6.989	6.814	7.157	7.000	4.090
Motor vehicle	6.881	6.446	7.131	7.250	6.173
Rain	6.243	5.928	6.306	6.750	5.738
Sneeze & cough	6.553	6.528	6.606	6.500	2.340
Average	6.967	6.657	7.154	7.214	5.540

Table 3: DCASE 2023 task 7 official results across all sound classes. **WAS** indicates “Weighted Average Score”, **Qual.** refers to audio quality, **Fit.** to category fitness, and **Div.** to diversity within the class.

Model	**WAS ↑**	**Qual. ↑**	**Fit. ↑**	**Div. ↑**	**FAD ↓**
Surrey	7.886	7.546	**8.419**	7.500	**3.621**
LINE	7.339	6.444	7.529	**8.750**	3.679
HEU	4.877	3.800	5.142	6.500	5.685
Baseline	2.688	2.930	2.447	-	13.412
Ours	**7.984**	**7.612**	8.223	8.250	11.456

Table 4: Comparison of the official results for the “Dog Bark” sound class in DCASE 2023 Task 7 with other submission models.

have observed that the diversity of the generated sound is not as sufficient as that of real sound samples from the training dataset. On the other hand, by employing a variety of text prompts for each class, our model is capable of generating a more diverse range of sounds. For example, for footstep sound class, we can provide prompts such as:“*clean* recording, footsteps on snow”, “*clean* recording, footsteps, running”, and “*clean* recording, footsteps in a large room”.

3. EVALUATION AND ANALYSIS

In DCASE 2023 Task 7, our model achieved 2nd place in subjective scores and 3rd place in FAD scores, with a specific breakdown of 2nd place in Audio Quality, 3rd place in Category Fit, and 1st place in Diversity. Table 3 presents the details of each sound class. In this section, we will delve deeply into the topics of objective and subjective evaluations.

The right column in Table 3 presents FAD scores across all sound classes using the official evaluation repositories.[6] Our approach outperforms the baseline approach in all classes, with notable improvements observed in the rain and moving motor vehicle classes. Furthermore, the subjective quality is significantly improved by our model in all classes. It should be acknowledged that FAD scores may not be indicative of other important aspects of audio quality such as clarity, high-SNR, and high-frequency components. Also, as FAD measures similarity between a reference set and a test set, improvement beyond reference is mismeasured as a degradation, including quantization noise and codec noise. As evidenced in Table 3 and Table 4, our performance in the “Dog Bark” sound class received the worst score in FAD, while achieving the highest score in the Weighted Average Score (WAS).

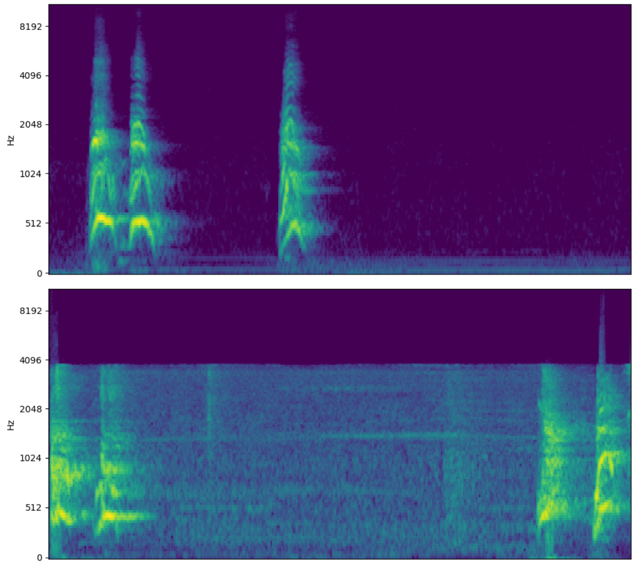

Figure 2: Mel-spectrograms of the generated audio samples using different recording environment prefixes. Prompts for images are (*top*) “*clean* recording, puppy bark,” and (*botton*) “*noisy* recording, puppy bark,” respectively

Our model was developed to generate high-quality audio suitable for real-world scenarios using the environment and audio quality prefixes. Despite most of the audio samples in our training dataset exhibiting poor audio quality due to background noise, babble noise, wind noise, device noise, and codec distortion, we confirmed our model produces high-quality audio. As discussed in Section 2.3, we controlled the audio sample quality by adding a special token as a prefix to the original text. Given that audio quality cannot be evaluated objectively, we conducted a informal listening test for the same text with both *clean* and *noisy* prefixes. Depending on the prefix used, we observed impressive improvements in sound quality across all sound classes. As illustrated in Figure 2, we can clearly observe that the use of the *clean* prefix had a discernible impact on the audio quality, as indicated by the mel-spectrogram images. This type of model steering by prompting has been popular in other domains, and to our best knowledge, our work is the first work that successfully shows it in audio generation.

To improve quality for mel-spectrogram inversion, we trained our own network based on HiFi-GAN [19] and BigV-

[6] https://github.com/DCASE2023-Task7-Foley-Sound-Synthesis

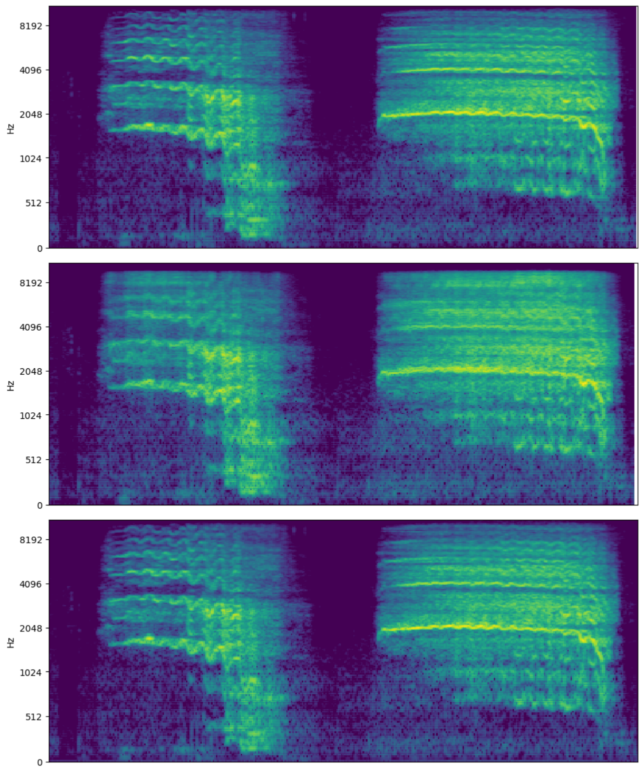

Figure 3: Mel-spectrogram for reconstructed samples. (*top*) ground-truth waveform, (*middle*) reconstructed with challenge baseline model, and (*bottom*) Our GOMIN system.

GAN [20] as explained above. Thanks to its bigger capacity and other architectural improvements, it showed better performance for overall sound categories. Compared to the baseline model,[6] our model well reconstructs tonal or harmonic components in the signal especially when the input mel-spectrograms include complex composition.

4. CONCLUSION

In this paper, we have presented FALL-E, Gaudio's foley synthesis system. FALL-E employs a cascaded approach with low-resolution spectrogram generation, a super-resolution module, and a vocoder. Our system was submitted to the DCASE 2023 Task 7 - Foley Synthesis Challenge (Track A), and we have reported the objective measure with respect to the official evaluation set. Through our extensive dataset and language model conditioning, as well as prompt engineering, we have achieved high-quality, diverse, and coherent sound generation results.

There is a vast potential for the development of generative AI in the audio domain. As technology continues to advance, new possibilities for sound generation arise, and the potential applications of this technology are vast. For example, in film and game production, foley synthesis could be used to produce more realistic sound effects, saving time and resources compared to traditional foley artistry. We believe that FALL-E, along with other works in the field, will pave the way for future advancements in generative audio technology, and we look forward to the continued development of this exciting area of research.

Acknowledgement

This research was supported by Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism in 2022 (Project Name: R&D on AI Text-to-Sound Generation, Project Number: RS-2023-00229204, Contribution Rate: 100%). The authors are grateful for Naver D2 Startup Factory and Naver Cloud Platform for supporting the GPU resources for this research.

We would like to highlight the clear arrangement implemented to ensure fairness and prevent any unfair advantage in the task. The conflict of interest with one of the organizers of this task was openly disclosed to the organizers, and the co-organizer affiliated with the institution in question remained uninvolved once the finalists were objectively determined. Additionally, during the subjective evaluation phase, other organizers were kept blind to the submission numbers to maintain impartiality. These measures were put in place to uphold the integrity and impartiality of the task evaluation process.

5. REFERENCES

[1] F. Kreuk, G. Synnaeve, A. Polyak, U. Singer, A. Défossez, J. Copet, D. Parikh, Y. Taigman, and Y. Adi, "Audiogen: Textually guided audio generation," *arXiv preprint arXiv:2209.15352*, 2022.

[2] H. Liu, Z. Chen, Y. Yuan, X. Mei, X. Liu, D. Mandic, W. Wang, and M. D. Plumbley, "Audioldm: Text-to-audio generation with latent diffusion models," 2023.

[3] 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.

[4] N. Zeghidour, A. Luebs, A. Omran, J. Skoglund, and M. Tagliasacchi, "Soundstream: An end-to-end neural audio codec," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 495–507, 2021.

[5] A. Défossez, J. Copet, G. Synnaeve, and Y. Adi, "High fidelity neural audio compression," *arXiv preprint arXiv:2210.13438*, 2022.

[6] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 684–10 695.

[7] L. Sheng, D.-Y. Huang, and E. N. Pavlovskiy, "High-quality speech synthesis using super-resolution mel-spectrogram," *arXiv preprint arXiv:1912.01167*, 2019.

[8] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 5485–5551, 2020.

[9] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[10] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[11] V. Liu and L. B. Chilton, "Design guidelines for prompt engineering text-to-image generative models," in *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, 2022, pp. 1–23.

[12] J. White, Q. Fu, S. Hays, M. Sandborn, C. Olea, H. Gilbert, A. Elnashar, J. Spencer-Smith, and D. C. Schmidt, "A prompt pattern catalog to enhance prompt engineering with chatgpt," *arXiv preprint arXiv:2302.11382*, 2023.

[13] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever, "Zero-shot text-to-image generation," 2021.

[14] T. A. Halgren, R. B. Murphy, R. A. Friesner, H. S. Beard, L. L. Frye, W. T. Pollard, and J. L. Banks, "Glide: a new approach for rapid, accurate docking and scoring. 2. enrichment factors in database screening," *Journal of medicinal chemistry*, vol. 47, no. 7, pp. 1750–1759, 2004.

[15] 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.

[16] J. Ho, C. Saharia, W. Chan, D. J. Fleet, M. Norouzi, and T. Salimans, "Cascaded diffusion models for high fidelity image generation," *The Journal of Machine Learning Research*, vol. 23, no. 1, pp. 2249–2281, 2022.

[17] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv:2204.06125*, 2022.

[18] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, A. Castro-Ros, M. Pellat, K. Robinson, D. Valter, S. Narang, G. Mishra, A. Yu, V. Zhao, Y. Huang, A. Dai, H. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, and J. Wei, "Scaling instruction-finetuned language models," 2022.

[19] 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.

[20] S.-g. Lee, W. Ping, B. Ginsburg, B. Catanzaro, and S. Yoon, "Bigvgan: A universal neural vocoder with large-scale training," *arXiv preprint arXiv:2206.04658*, 2022.

[21] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville, "Film: Visual reasoning with a general conditioning layer," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[22] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 776–780.

[23] K. Drossos, S. Lipping, and T. Virtanen, "Clotho: An audio captioning dataset," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 736–740.

[24] 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.