A DUAL-PATH FRAMEWORK WITH FREQUENCY-AND-TIME EXCITED NETWORK FOR ANOMALOUS SOUND DETECTION

Yucong Zhang[1,2] *Juan Liu*[1†] *Yao Tian*[3] *Haifeng Liu*[4] *Ming Li*[1,2†]

[1]School of Computer Science, Wuhan University, Wuhan, China
[2]Suzhou Municipal Key Laboratory of Multimodal Intelligent Systems, Duke Kunshan University, Kunshan, China
[3]Data & AI Engineering System, OPPO, Beijing, China
[4]University of Science and Technology of China, Hefei, China

ABSTRACT

In contrast to human speech, machine-generated sounds of the same type often exhibit consistent frequency characteristics and discernible temporal periodicity. However, leveraging these dual attributes in anomaly detection remains relatively under-explored. In this paper, we propose an automated dual-path framework that learns prominent frequency and temporal patterns for diverse machine types. One pathway uses a novel Frequency-and-Time Excited Network (FTE-Net) to learn the salient features across frequency and time axes of the spectrogram. It incorporates a Frequency-and-Time Chunkwise Encoder (FTC-Encoder) and an excitation network. The other pathway uses a 1D convolutional network for utterance-level spectrum. Experimental results on the DCASE 2023 task 2 dataset show the state-of-the-art performance of our proposed method. Moreover, visualizations of the intermediate feature maps in the excitation network are provided to illustrate the effectiveness of our method.

***Index Terms*—** Anomalous sound detection, squeeze and excitation, frequency pattern analysis, temporal periodicity analysis

1. INTRODUCTION

Anomalous sound detection (ASD) is a task to distinguish anomalous sounds from normal ones. It is useful to monitor a machine's condition and detect malfunctions of an operating machine before it is damaged. ASD is a challenging task and is often regarded as an unsupervised learning problem [1], given the rare occurrence and high diversity of anomalous events. Furthermore, in real-world scenarios, machines may operate under different settings and environmental conditions, leading to potential domain shifts [2–5], thereby increasing the difficulty of the ASD task.

To address the lack of anomalous data, conventional ASD systems adopt a generative method [6, 7] to model the distribution of normal data. Recently, self-supervised methods [8–11] are getting more attention, which is widely adopted by top-ranked teams [12–18] in recent DCASE[1] challenges. These systems train a feature extractor on normal data to obtain expressive embeddings, and use distance metrics to assess the abnormality by comparing test embeddings with normal ones. Despite the success of these systems, the frequency patterns and temporal periodicity remain relatively under-explored when modeling machine sounds.

Some recent studies have investigated the efficacy of frequency patterns in machine-generated sounds. In DCASE 2022 Challenge, the first-ranking team [12] builds customized high-pass filters for individual machine types, enhancing ASD performance by applying them before the Mel filters. Additionally, experiments conducted by [19] demonstrate notable high-frequency characteristics produced by certain machine types. Nevertheless, these approaches rely on manually constructed filters to leverage frequency patterns, limiting their adaptability to new machine types.

To automatically explore the frequency patterns, one possible solution is to learn the patterns with deep learning. Recently, researchers in [20] have explored automated analysis of frequency patterns on top of their prior work [10]. They introduce a multi-head self-attention [21] to adaptively filter the log-Mel spectrogram. Their experimental results demonstrate the feasibility of integrating frequency pattern analysis into the training process of ASD.

In this paper, we propose a novel framework that leverages both the frequency and temporal characteristics. We use the framework from [22] as the backbone, dealing with both frame-level spectrogram and utterance-level spectrum. Different from [22], we employ a Frequency-and-Time Excited Network (FTE-Net) in the spectrogram pathway to enrich the learnt representation by capturing salient patterns in both the frequency and time domains. To the best of our knowledge, our work is the first to integrate both frequency and temporal pattern analysis of a spectrogram within a deep-learning

†Corresponding Authors: Juan Liu: liujuan@whu.edu.cn, Ming Li: ming.li369@dukekunshan.edu.cn

[1]DCASE: Detection and Classification of Acoustic Scenes and Events, https://dcase.community

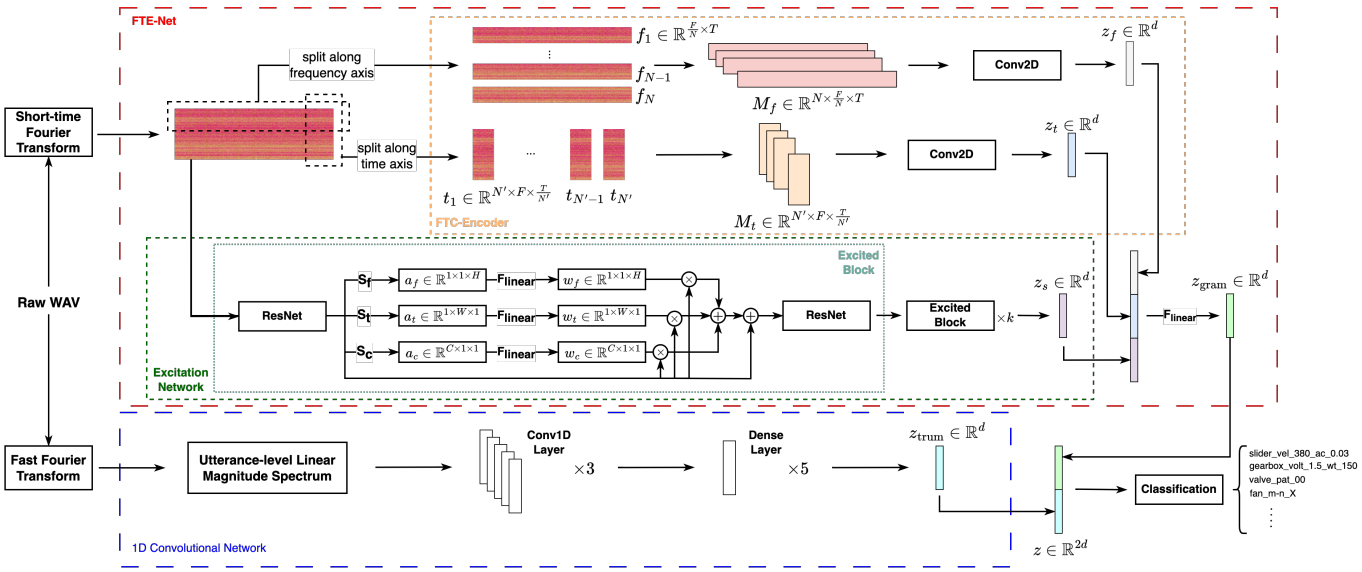


Fig. 1: The overview of our proposed framework

framework for machine ASD.

2. METHODS

Our proposed framework uses [22] as the backbone, integrating a 1D convolutional network for utterance-level spectrum and an FTE-Net for frame-level spectrogram. The FTE-Net incorporates a Frequency-and-Time Chunkwise Encoder (FTC-Encoder) and an excitation network. The overall structure of our method is depicted in Fig. 1. In section 2.1, we introduce the backbone framework [22] and briefly explain the difference between theirs [22] and ours. In section 2.2, we introduce the proposed FTE-Net module and explain in detail the FTC-Encoder and the excitation network in the module.

2.1. Backbone framework

The backbone framework is a dual-path ASD framework [22], designed to process both the frame-level spectrogram and utterance-level spectrum of machine-generated sounds in separate paths. The spectrum is processed by three 1D convolutional layers and five dense layers, and the spectrogram is processed by four ResNet [23] layers. Comparing to using only the spectrogram, empirical results from top-ranked teams [15, 18] show that by adopting such dual-path structure that handles the spectrogram and spectrum separately can produce better results. In this work, we replace the network used in the spectrogram pathway of [22] with a novel FTE-Net, aiming to learn frequency and temporal patterns.

2.2. Frequency-and-Time Excited Network (FTE-Net)

The FTE-Net is a two-branch network. One branch employs an FTC-Encoder, and the other branch uses an excitation network. The FTC-Encoder allows the network to learn the potential patterns within small intervals of frequency or time, while the excitation network is used to filter out unrelated information and enhance the useful patterns in a global context.

2.2.1. *Frequency-and-Time Chunkwise Encoder (FTC-Encoder)*

Table 1: Structure of the Conv2D module in the FTC-Encoder. n indicates the number of layers or blocks, c is the number of output channels, k is the kernel size and s is the stride. h and w are the output height and width of the ResNet Blocks.

Operator	n	c	k	s
Conv2D 7x7	1	32	(7,7)	(2,2)
MaxPooling	-	-	(3,3)	(2,2)
ResNet block	4	(64, 128, 128, 128)	(3,3)	(2,2)
MaxPooling	-	-	(h, w)	(h, w)

The FTC-Encoder is designed to process spectrogram data in a chunkwise manner, with separate pathways dedicated to handle frequency chunks and time chunks respectively. The goal of this module is to capture potential patterns within short frequency bands and time intervals.

In the frequency pathway, the input spectrogram $X \in \mathbb{R}^{F\times T}$ is equally segmented into N overlapping frequency bands, denoted as $f_i \in \mathbb{R}^{\frac{F}{N}\times T}$. These frequency bands $f_1, f_2, \cdots, f_N$ are subsequently merged to create a band-wise 3D feature matrix $M_f \in \mathbb{R}^{N\times\frac{F}{N}\times T}$. Finally, M_f is passed through a 2D convolution network (as shown in Table 1) to get the embedding $z_f \in \mathbb{R}^d$, with the number of chunks serving as the number of input channels. The first Conv2D and MaxPooling layer uses large kernel size, aiming to reduce the dimension of the input. The last MaxPooling layer is used to flatten the feature maps.

Similar strategies are applied to the dual pathway along the time axis, using the same structure after splitting the spectrogram into small time segments.

2.2.2. *Excitation network*

The detailed structure of the excitation network is shown in Table 2. Modified squeeze-and-excitation (SE) [24] mod-

Table 2: Structure of the excitation network with the same notations shown in Table. 1.

Operator	n	c	k	s
Modified SE	-	-	-	-
Conv2d	1	16	(7,7)	(2,2)
MaxPooling	-	-	(3,3)	(2,2)
Modified SE	-	-	-	-
ResNet Block		16	(3,3)	(1,1)
Modified SE	1	-	-	-
ResNet Block		16	(3,3)	(1,1)
ResNet Block		(32, 64, 128, 256)	(3,3)	(2,2)
Modified SE	4	-	-	-
ResNet Block		(32, 64, 128, 256)	(3,3)	(1,1)
MaxPooling	-	-	(h, w)	(h, w)

ules are integrated between ResNet blocks to form the excited block. While the conventional SE generates a mask (w_c) to adjust channel-wise feature maps, we introduce two additional masks, namely the frequency excitation mask (w_f) and the time excitation mask (w_t). As shown in Fig. 1, given an input $x \in \mathbb{R}^{C\times H\times W}$, where H and W are the dimensions along the frequency and time axis, the excitation map is formulated as follows:

$$w_i = \frac{1}{1+\exp\big(-(a_i \cdot W^{\mathrm{T}} + b)\big)}, \quad a_i = S_i(x)$$

where S_i is a 2D average pooling operation, cancelling out the dimension other than i. W and b are learning parameters. The output is aggregated using the excitation maps as follows:

$$y = x + \sum_{i\in\{c,f,t\}} w_i(x)\cdot x,$$

where w_c, w_f, and w_t represent the excitation masks for channel, frequency, and time respectively.

As a result, the output embeddings (z_f, z_t) of the FTC-Encoder, the embedding (z_s) of the excitation network are concatenated before passing to a linear layer to get the spectrogram embedding (z_{gram}). Meanwhile, the spectrum embedding (z_{trum}) is generated by the 1D convolutional network. To train the embeddings, z_{gram} and z_{trum} are stacked together, used as an input to a linear classifier to classify different machines.

3. EXPERIMENTS

3.1. Dataset

The experiments are conducted on the DCASE 2023 Task 2 dataset [3], which comprises audio clips from seven distinct machine types. Each machine type has roughly 1,000 audio clips, including 990 clips of source data and 10 clips of target data. Each audio clip lasts 6 to 18 seconds with a sampling rate of 16 kHz. The dataset includes a development dataset, an additional dataset, and an evaluation dataset. To compare with other systems in the challenge, the model is trained using the training portion of the development dataset and the additional dataset, while performance evaluation is conducted on the evaluation dataset. It is important to note that only normal machine sounds are used for training.

Table 3: Results (%) on DCASE 2023 task 2 evaluation dataset. source AUC, target AUC, mean AUC and pAUC is the harmonic mean over all machine types. Integrated score is calculated using the official script[2].

System	source AUC	target AUC	mean AUC	pAUC	Integrated Score
Official baseline [28]	-	-	63.41	56.82	61.05
Jie et al. [15]	-	-	69.75	62.03	66.97
Lv et al. [16]	-	-	70.04	60.01	66.39
Jiang et al. [17]	-	-	68.03	60.71	65.40
Wilkinghoff [18]	-	-	67.95	59.58	64.91
Self-implement Baseline	**76.31**	66.72	72.34	62.91	68.20
Proposed Method using FTE-Net	72.94	**75.08**	**73.97**	**66.38**	**71.27**

3.2. Implementation details

For data processing, we use linear magnitude spectrograms and spectrum as the inputs. The spectrogram is obtained by Short-time Fourier Transform, with the sampling window size and hop length set to 1024 and 512 respectively. The entire signal is used to obtain the utterance-level spectrum. In our experiments, we repeat and clip the audio to force its length to be 18 seconds.

In terms of the training strategy, we use the sub-cluster AdaCos [25] as the loss function to train the model. Wave-level mixup [26] strategy is adopted as the data augmentation. We set the number of classes to match the joint categories of machine types and attributes. The model is optimized with the ADAM optimizer [27] with a learning rate of 0.001. We set the batch size to 64 and train the model for 100 epochs.

The ASD results are generated by measuring the cosine distance between the prototypes of normal embeddings with the test embeddings for each machine type. Each machine type has 26 prototypes, including 16 center embeddings generated by K-Means on the source domain, and all the 10 embeddings from the target domain.

The results are evaluated using the official scripts[2]. Three commonly used metrics are adopted for evaluating the ASD performance in this paper: AUC, pAUC and the integrated scores. AUC is divided into source AUC and target AUC for the data in separate domains. pAUC is calculated as the AUC over a low false-positive-rate (FPR) range [0, 0.1]. The integrated score is the harmonic mean of AUC and pAUC across all machine types, which is the official score used for ranking.

3.3. Performance comparison and ablation studies

We compare the performance of our proposed framework with the top 4 teams [15–18] in the DCASE 2023 challenge. As presented in Table 3, our method surpasses all teams across all evaluation metrics. Notably, our approach exhibits a superior performance with a 4.3% absolute improvement over the first-ranking team [15] and a substantial 10.22% absolute improvement over the official system [28] in terms

[2]Official scripts available at `https://github.com/nttcslab/dcase2023_task2_evaluator`

of the integrated score. The self-implement baseline shown in the table is re-implementation of [18] with more ResNet blocks added to the spectrogram branch. The results indicate that the proposed FTE-Net leads to improvements in the overall ASD performance.

Moreover, we find that the proposed framework exhibits a noteworthy capacity for domain generalization. As observed in Table 3, despite a moderate reduction in the source AUC compared to the baseline system, our framework demonstrates a substantial improvement in terms of the target AUC. We argue that the inferior performance of the source AUC is likely attributed to overfitting of the baseline system to the source data, given that the source and target domains feature a highly imbalanced ratio. An indicator of the overfitting phenomenon in the baseline system is the significant disparity between the source and target AUC values presented in the table. In contrast, the proposed FTE-Net exhibits a relatively minor difference, showing its generalization ability.

Table 4: Results (%) for different modules in FTE-Net.

System	mean AUC	pAUC	Integrated Score
FTE-Net	**73.97**	**66.38**	**71.27**
w/o FTC-Encoder	70.46	65.08	68.58
w/o Excitation Network	71.78	64.18	69.06
w/o Both (Self-implement Baseline)	72.34	62.91	68.20

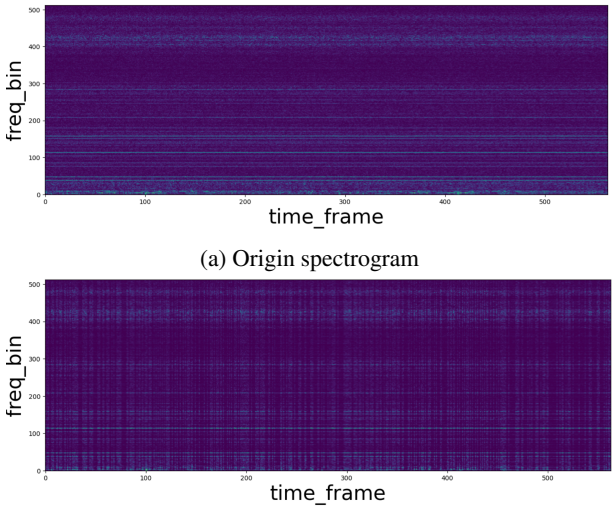
Table 5: Results (%) using different excitation mechanism.

System	mean AUC	pAUC	Integrated Score
FTE-Net	**73.97**	**66.38**	**71.27**
w/o Both (Vanilla SE)	69.94	62.31	67.20
w/o freq. excitation	69.43	64.48	67.79
w/o time excitation	70.45	63.84	68.10

To show the effectiveness of the individual modules in FTE-Net, we conduct some ablation studies. In Table 4, we show that the best performance is achieved by using all the modules. In Table 5, we conduct an excitation mechanism ablation study. Our findings demonstrate that employing more excitation maps results in improved performance. Notably, frequency excitation maps outperform time excitation maps in terms of ASD performance.

3.4. Visualization analysis

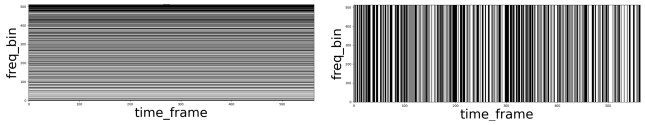
To illustrate the impact of excitation mechanism in the excitation network, we present spectrogram comparisons before and after applying the excitation maps. In this example featuring a fan shown in Fig. 2, we observe that the original spectrogram undergoes enhancement both in terms of frequency and time, highlighting the effectiveness of our method. Particularly, in the frequency excitation map shown in Fig. 3 (a), our network predominantly focuses on the high-frequency band, in accordance with the results given by recent discoveries [12, 19, 20]. This indicates that our method effectively generates excitation maps conducive to machine sound modeling.



(a) Origin spectrogram

(b) Feature map after the first excitation

Fig. 2: Illustration of the excited mechanism on fan



(a) frequency-wise (b) time-wise

Fig. 3: Illustration of excitation masks on fan

From Fig. 3 (a) and (b), frequency and temporal patterns can be highlighted. For example, despite the enhancement of the high frequency, some prominent frequency patterns in the middle range of the spectrogram are highlighted while some of them are filtered out. Additionally, despite the simple temporal periodicity, much more complicated temporal patterns within tiny time segments are shown. These patterns hold potential as features for analyzing sounds emitted by specific machine types in future research.

4. CONCLUSION

In our paper, we introduce a novel dual-path framework for anomaly detection in machine-generated sounds, which has the ability to leverage distinctive frequency and temporal patterns found in machine sounds. One pathway employs the Frequency-and-Time Excited Network (FTE-Net) to capture features across both frequency and time axes of the spectrogram. The other pathway utilizes a 1D convolutional network for utterance-level spectrum. The experiments on the DCASE 2023 task 2 dataset shows that our framework achieves state-of-the-art performance, demonstrating the effectiveness of leveraging dual attributes for machine ASD.

5. ACKNOWLEDGEMENTS

This research is funded in part by the Science and Technology Program of Suzhou City (SYC2022051) and National Natural Science Foundation of China (62171207). Many thanks for the computational resource provided by the Advanced Computing East China Sub-Center.

6. REFERENCES

[1] Y. Koizumi, Y. Kawaguchi, K. Imoto, T. Nakamura, Y. Nikaido, R. Tanabe, H. Purohit, K. Suefusa, T. Endo, M. Yasuda, and N. Harada, "Description and discussion on dcase2020 challenge task2: Unsupervised anomalous sound detection for machine condition monitoring," in *Proc. of DCASE 2020 Workshop*, 2020.

[2] K. Dohi, K. Imoto, N. Harada, D. Niizumi, Y. Koizumi, T. Nishida, H. Purohit, R. Tanabe, T. Endo, M. Yamamoto, and Y. Kawaguchi, "Description and discussion on DCASE 2022 challenge task 2: Unsupervised anomalous sound detection for machine condition monitoring applying domain generalization techniques," in *Proc. of DCASE 2022 Workshop*, 2022.

[3] K. Dohi, K. Imoto, N. Harada, D. Niizumi, Y. Koizumi, T. Nishida, H. Purohit, R. Tanabe, T. Endo, and Y. Kawaguchi, "Description and discussion on dcase 2023 challenge task 2: First-shot unsupervised anomalous sound detection for machine condition monitoring," *ArXiv*, vol. abs/2305.07828, 2023.

[4] N. Harada, D. Niizumi, D. Takeuchi, Y. Ohishi, M. Yasuda, and S. Saito, "Toyadmos2: Another dataset of miniature-machine operating sounds for anomalous sound detection under domain shift conditions," in *Proc. of DCASE 2021 Workshop*, 2021, pp. 1–5.

[5] K. Dohi, T. Nishida, H. Purohit, R. Tanabe, T. Endo, M. Yamamoto, Y. Nikaido, and Y. Kawaguchi, "MIMII DG: sound dataset for malfunctioning industrial machine investigation and inspection for domain ggeneralization task," in *Proc. of DCASE 2022 Workshop*, 2022.

[6] E. Rushe and B. M. Namee, "Anomaly detection in raw audio using deep autoregressive networks," in *Proc. of ICASSP*, 2019, pp. 3597–3601.

[7] K. Suefusa, T. Nishida, H. Purohit, R. Tanabe, T. Endo, and Y. Kawaguchi, "Anomalous sound detection based on interpolation deep neural network," in *Proc. of ICASSP*, 2020, pp. 271–275.

[8] H. Hojjati and N. Armanfard, "Self-supervised acoustic anomaly detection via contrastive learning," in *Proc. of ICASSP*, 2022, pp. 3253–3257.

[9] H. Chen, Y. Song, L.-R. Dai, I. McLoughlin, and L. Liu, "Self-supervised representation learning for unsupervised anomalous sound detection under domain shift," in *Proc. of ICASSP*, 2022, pp. 471–475.

[10] Y. Liu, J. Guan, Q. Zhu, and W. Wang, "Anomalous sound detection using spectral-temporal information fusion," in *Proc. of ICASSP*, 2022, pp. 816–820.

[11] Y. Zhang, S. Hongbin, Y. Wan, and M. Li, "Outlier-aware Inlier Modeling and Multi-scale Scoring for Anomalous Sound Detection via Multitask Learning," in *Proc. of INTERSPEECH*, 2023, pp. 5381–5385.

[12] Y. Zeng, H. Liu, L. Xu, Y. Zhou, and L. Gan, "Robust anomaly sound detection framework for machine condition monitoring," DCASE 2022 Challenge, Tech. Rep., July 2022.

[13] I. Kuroyanagi, T. Hayashi, K. Takeda, and T. Toda, "Two-stage anomalous sound detection systems using domain generalization and specialization techniques," DCASE 2022 Challenge, Tech. Rep., July 2022.

[14] F. Xiao, Y. Liu, Y. Wei, J. Guan, Q. Zhu, T. Zheng, and J. Han, "The dcase2022 challenge task 2 system: Anomalous sound detection with self-supervised attribute classification and gmm-based clustering," DCASE 2022 Challenge, Tech. Rep., July 2022.

[15] J. Jie, "Anomalous sound detection based on self-supervised learning," DCASE 2023 Challenge, Tech. Rep., June 2023.

[16] Z. Lv, B. Han, Z. Chen, Y. Qian, J. Ding, and J. Liu, "Unsupervised anomalous detection based on unsupervised pretrained models," DCASE 2023 Challenge, Tech. Rep., June 2023.

[17] A. Jiang, Q. Hou, J. Liu, P. Fan, J. Ma, C. Lu, Y. Zhai, Y. Deng, and W.-Q. Zhang, "Thuee system for first-shot unsupervised anomalous sound detection for machine condition monitoring," DCASE 2023 Challenge, Tech. Rep., June 2023.

[18] K. Wilkinghoff, "Fraunhofer fkie submission for task 2: First-shot unsupervised anomalous sound detection for machine condition monitoring," DCASE 2023 Challenge, Tech. Rep., June 2023.

[19] K. T. Mai, T. Davies, L. D. Griffin, and E. Benetos, "Explaining the decision of anomalous sound detectors," in *Proc. of DCASE 2022 Workshop*, 2022.

[20] H. Zhang, J. Guan, Q. Zhu, F. Xiao, and Y. Liu, "Anomalous Sound Detection Using Self-Attention-Based Frequency Pattern Analysis of Machine Sounds," in *Proc. INTERSPEECH*, 2023, pp. 336–340.

[21] A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. of NIPS*, 2017.

[22] K. Wilkinghoff, "Design choices for learning embeddings from auxiliary tasks for domain generalization in anomalous sound detection," *Proc. of ICASSP*, 2023.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. of CVPR*, 2016, pp. 770–778.

[24] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. of CVPR*, 2018, pp. 7132–7141.

[25] K. Wilkinghoff, "Sub-cluster AdaCos: Learning representations for anomalous sound detection," in *Proc. of IJCNN*, 2021.

[26] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proc. of ICLR*, 2018.

[27] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.

[28] N. Harada, D. Niizumi, Y. Ohishi, D. Takeuchi, and M. Yasuda, "First-shot anomaly sound detection for machine condition monitoring: A domain generalization baseline," *ArXiv*, vol. abs/2303.00455, 2023.