Multi-scale Scanning Network for Machine Anomalous Sound Detection

Yucong Zhang$^{1,2[0009-0001-6553-3890]}$, Juan Liu$^{1,3\star[0000-0001-9344-7415]}$, and
Ming Li$^{1,2\star[0000-0002-6406-1983]}$

[1] School of Computer Science, Wuhan University, Wuhan, China
[2] Suzhou Municipal Key Laboratory of Multimodal Intelligent Systems, Digital Innovation Research Center, Duke Kunshan University, Suzhou, China
[3] School of Artificial Intelligence, Wuhan University, Wuhan, China
{yucong.zhang, ming.li369}@dukekunshan.edu.cn

Abstract. Machine sounds exhibit consistent and repetitive patterns in both the frequency and time domains, which vary significantly across scales for different machine types. For instance, rotating machines often show periodic features in short time intervals, while reciprocating machines exhibit broader patterns spanning the time domain. While prior studies have leveraged these patterns to improve Anomalous Sound Detection (ASD), the variation of patterns across scales remains insufficiently explored. To address this gap, we introduce a Multi-scale Scanning Network (MSN) designed to capture patterns at multiple scales. MSN employs kernel boxes of varying sizes to scan audio spectrograms and integrates a lightweight convolutional network with shared weights for efficient and scalable feature representation. Experimental evaluations on the DCASE 2020 and DCASE 2023 Task 2 datasets demonstrate that MSN achieves state-of-the-art performance, highlighting its effectiveness in advancing ASD systems.

Keywords: Anomalous sound detection · Multi-scale · Representation learning

1 Introduction

Machine Anomalous Sound Detection (ASD) aims to differentiate abnormal machine operating sounds from normal ones. Due to the scarcity of anomalies, ASD tasks often require models to detect abnormal samples without prior exposure to them [16, 6]. In recent years, a variety of methods have been developed for the ASD task. Several generative methods have been found useful, aiming to model the distribution of the normal data by reconstructing audio spectrograms [22, 23]. However, their strong generalization capability can lead to the unintended reconstruction of anomalous samples [18, 29], resulting in detection failures. To overcome this limitation, Discriminative Representation Learning (DRL) methods [8, 35, 27] have gained prominence. These approaches learn robust representations by classifying audio clips based on supplementary information like machine

[⋆] Corresponding authors: Juan Liu and Ming Li types, and have proven highly effective in recent DCASE challenges. However, they often require large amounts of annotated data, which can be difficult to collect or annotate.

To address the challenge of limited training samples in ASD, researchers have explored data augmentation and fine-tuning strategies to enhance DRL models. Data augmentation involves generating synthetic samples by introducing anomalies on audio spectrograms [29, 1, 24], creating fake samples in latent spaces [27, 30], or simulating machine sounds with diverse physical properties through finite element analysis [37]. Pre-trained generative models like AudioLDM [19] have also been used to generate machine sounds under varying conditions by translating operational attributes into textual descriptions [33]. While these methods diversify training data, poorly designed synthetic samples risk degrading model performance.

Fine-tuning pre-trained models has emerged as a promising solution for few-shot ASD tasks. Recent studies demonstrate the effectiveness of contrastive learning in initializing DRL model weights using in-domain machine data, providing a robust starting point. A discriminative task is then employed to train the representation specifically for ASD [10]. Notably, transferring weights pre-trained on large-scale speech data to ASD yields competitive results after fine-tuning on machine audio [11]. To better align pre-trained models with the inductive bias of machine audio, researchers have fine-tuned models like BEATs [2] and CED [5], originally trained on large-scale datasets such as AudioSet [7]. This approach significantly enhances ASD task performance, achieving state-of-the-art (SOTA) results [15, 38]. However, the fixed transformer-based architectures of these methods limit flexibility, posing challenges for adaptation and customization in ASD tasks. This limitation is particularly significant given the distinct spectrogram patterns identified in previous studies, which have proven promising for anomaly detection [31, 21].

To automate the exploration of these spectrogram patterns, a range of methods has been proposed. For example, the multi-head self-attention mechanism [25] has been employed to adaptively filter log-Mel spectrograms [32]. Similarly, global weighted ranking pooling (GWRP) [17] has been applied to the time domain of spectrograms [9], adapting to different machine types. Additionally, squeeze-and-excitation modules and band-wise splitting strategy have been investigated to capture both temporal and spectral patterns during model training [36]. The experimental results of all these approaches demonstrate the effectiveness of incorporating spectrogram pattern analysis into the ASD training processes.

While several studies have explored automatic feature extraction from machine spectrograms, few have addressed patterns across multiple scales. To address this gap, we propose a Multi-scale Scanning Network (MSN) that utilizes multiple kernel boxes with different sizes to capture information across the entire spectrogram. The outputs from all kernel boxes are processed through a shared ResNet-based network [12], and the resulting features are concatenated into a unified embedding, which is fed into auxiliary classification layers. By leverag-

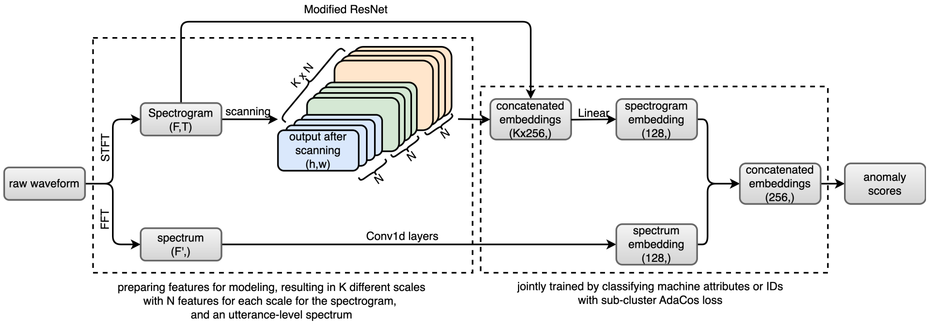
Modified ResNet
raw waveform
STFT
FFT
Spectrogram (F,T)
scanning
K x N
output after scanning (h,w)
N
N
N
spectrum (F',)
Conv1d layers
concatenated embeddings (Kx256,)
Linear
spectrogram embedding (128,)
spectrum embedding (128,)
concatenated embeddings (256,)
anomaly scores
preparing features for modeling, resulting in K different scales with N features for each scale for the spectrogram, and an utterance-level spectrum
jointly trained by classifying machine attributes or IDs with sub-cluster AdaCos loss

Fig. 1. The overview of our proposed method. K is the number of kernel boxes used. N is the total number of features scanned by each kernel. F,T and F' are the dimensions of the spectrogram and spectrum computed from the input audio clip.

ing this multi-scale approach, our method effectively learns diverse patterns and enhances the representation of machine sound features. We evaluate our model on the DCASE 2020 and DCASE 2023 Task 2 benchmarks, demonstrating superior performance with the proposed module. Our implementation is publicly available on GitHub[4].

2 Proposed Method

2.1 Backbone

Our method employs the widely-used dual-path structure as its backbone, as shown in Figure 1. Each path consists of a sub-network, and their outputs are concatenated to generate a unified embedding. The sub-network lying below in the figure processes the utterance-level spectrum, capturing magnitude information across the entire frequency range of the spectrogram. The second sub-network utilizes the magnitude spectrogram, preserving frequency information over time. This dual-path architecture has demonstrated strong performance in ASD tasks [27, 36, 20]. Its effectiveness may stem from the spectrum's ability to complement information potentially missing in the spectrogram, while high-frequency resolution proves essential for certain machine types.

2.2 Spectrogram Encoding

As depicted in the top section of Figure 1, the raw waveform is converted into an audio spectrogram using the Short Time Fourier Transform (STFT). This transformation captures both time and frequency information, emphasizing local variations and characteristics within the audio signal. A modified ResNet [36] architecture is employed to process the spectrogram, as detailed in Table 1. The

[4] Codes available at https://github.com/yucongzh/MSN-Net

Table 1. Structure of the modified ResNet block shown in Figure 1. n indicates the number of layers or blocks, c is the number of output channel or dimension, k is the kernel size and s is the stride. This is used to encode the audio spectrogram.

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

Table 2. Structure of the Conv1d layers shown in Figure 1 with the same notations shown in Table 1. This is used to encode the utterance-level spectrum.

Operator	n	c	k	s
Conv1d	1	128	256	64
Conv1d	1	128	64	32
Conv1d	1	128	32	4
flatten	-	-	-	-
Linear	5	128	-	-

Table 3. Structure of the convolution module for the multi-scale inputs with the same notations shown in Table 1. This module convert the multi-scale features into embeddings with the same dimension.

Operator	n	c	k	s
ResNet block	2	(32, 64)	(3,3)	(2,2)
ResNet block	1	64	(3,3)	(1,1)
StatsPool	1	-	-	-
Linear	1	1024	-	-
Linear	1	256	-	-

modified ResNet integrates enhanced Squeeze-and-Excitation (SE) [14] modules, which assign dynamic weights not only across different channels but also along other dimensions. This design enables the model to effectively capture features across both the frequency and time axes, improving its representational capacity.

2.3 Spectrum Encoding

As illustrated in the lower section of Figure 1, the raw waveform is transformed into an utterance-level spectrum using Fast Fourier Transform (FFT). This approach captures the overall frequency content of the audio signal, ensuring comprehensive representation of the spectral information. Following established methods for spectrum encoding [27], the spectrum undergoes processing through several 1D convolutional layers, followed by fully connected linear layers to derive the final feature representation. The detailed architecture of the spectrum encoding pathway is outlined in Table 2.

2.4 Multi-scale Scanning Network

In order to capture local features in the spectrogram of machine sounds, we design small multi-scale kernel boxes that scans the whole spectrogram along time and frequency domains, and put those scanned features together for modeling. The whole pipeline includes multi-scale scanning, and representation learning.

Multi-scale Scanning We employ a collection of kernel boxes with varying dimensions, denoted as $\{K_{h\times w}\}$, where h and w represent the height and width of the kernel boxes, respectively. To simplify implementation, h and w are selected from powers of 2, ensuring computational efficiency and scalability across varying spectrogram resolutions. To emphasize frequency patterns, we use rectangular kernel boxes, with h generally larger than w, to capture frequency-dominant features while maintaining time-domain granularity, motivated by the observation that machine sounds often exhibit distinct frequency characteristics Specifically, we define $K = 12$ kernel boxes $\{K_{h\times w}\}$, with $h \in \{32, 64, 128, 256\}$ and $w \in \{16, 32, 64\}$. This diversity enables the model to focus on patterns of varying scales, effectively capturing both fine- and coarse-grained features within the spectrogram.

For each kernel box, N_f scans are performed along the frequency axis and N_t scans along the time axis, producing $N = N_f \times N_t$ features, each with fixed dimensions (h, w). The scanning process is designed to ensure full coverage of the spectrogram by employing a calculated hop length, which determines the step size for sliding the kernel box. The hop length is calculated to balance coverage and computational efficiency, ensuring that no regions of the spectrogram are left unexamined while avoiding redundant computations. For kernel boxes of different scales, the step sizes for frequency (F_step) and time (T_step) are computed as follows. Given an input spectrogram of dimensions (F, T) and a kernel box of size (h, w), the step sizes F_step and T_step are defined as:

$$\text{F\_step} = \begin{cases} \max\left(1, \left\lfloor \frac{F-h}{N_f-1} \right\rfloor\right) & \text{if } N_f > 1, \\ F - h & \text{otherwise.} \end{cases}$$

$$\text{T\_step} = \begin{cases} \max\left(1, \left\lfloor \frac{T-w}{N_t-1} \right\rfloor\right) & \text{if } N_t > 1, \\ T - w & \text{otherwise.} \end{cases}$$

The calculated step sizes dynamically adapt to the spectrogram dimensions and kernel box configurations, enabling the extraction of features that reflect the underlying spectral structure at each scale. This adaptability is crucial for handling spectrograms of varying resolutions and ensuring consistent feature extraction across different datasets. Each kernel box $K_{h\times w}$ extracts N features from the input spectrogram. These features are stacked to form an N-channel feature map of dimensions (N, h, w), as illustrated in Figure 1. The multi-scale structure allows the model to aggregate features from varying resolutions, enhancing its ability to capture localized and global patterns.

The output features from each kernel box are subsequently processed by a lightweight convolutional network. This network is specifically designed to extract scale-specific information while minimizing computational overhead, ensuring that the embedding captures essential spectral details without introducing significant latency. By integrating features from multiple scales, the model achieves a comprehensive representation of the input spectrogram, which is critical for downstream tasks such as classification or anomaly detection.

Representation Learning A lightweight convolutional network integrates outputs from multi-scale scanning into a unified embedding, enabling efficient and scalable feature representation. The network, detailed in Table 3, begins with an input of size (N, h, w), which is transformed into a lower-dimensional space through two residual blocks [12]. These blocks preserve critical spatial and contextual information while reducing dimensionality, leveraging skip connections to mitigate gradient vanishing and ensure stable training. A statistical pooling layer then computes channel-wise mean and standard deviation, producing a compact $(256,)$ embedding that captures robust, transformation-invariant statistics. If scanning is performed with K multi-scale kernel boxes, K embeddings are generated, each encapsulating information specific to a distinct scale. These embeddings are concatenated into a super embedding of size $(K \times 256,)$, which is refined by linear layers to produce the final spectrogram embedding, as visualized in Figure 1.

2.5 Anomaly Detection

Anomaly scores are calculated as the minimum cosine distance between prototypes of normal embeddings from the training dataset and test embeddings. For both DCASE 2020 and DCASE 2023 datasets, the same method is applied with different configurations. For DCASE 2020, scores are computed for each machine ID and type by comparing normal training samples with test samples of the same ID and type. For DCASE 2023, scores are computed for each machine type independently. Prototypes for each category are generated using K-Means clustering. For DCASE 2023, prototypes are created for both source and target domains, and the minimum cosine distance between the test sample and these prototypes is selected as the final anomaly score.

3 Experiments

3.1 Datasets

The experiments were conducted using the Task 2 datasets from the DCASE 2020 and DCASE 2023 challenges [16, 6]. Both datasets feature a development dataset and an evaluation dataset, each containing a training subset with only normal audio clips and a test subset with both normal and anomalous audio clips. These datasets are widely recognized in the ASD community, with DCASE 2020 focusing on standard ASD tasks and DCASE 2023 addressing ASD under domain shifts. The DCASE 2020 dataset includes six machine types, each with multiple machine IDs, while the DCASE 2023 dataset comprises 14 machine types with data from both source and target domains. The DCASE 2023 dataset also

Table 4. Model Comparison on the development test dataset of the DCASE 2020 Task 2 dataset. All results (%) are reported in terms of the mean of the AUC and pAUC. All the models are trained on only the normal machine sounds of the training dataset. "-" means that the result is not reported in the source paper.

DCASE 2020 Task 2								
	Pre-	Development Dataset						
Models	-trained	Fan	Pump	Slider	T.Car	T.Conv	Valve	Mean
2020 No.1 [8]	-	80.65	83.27	93.41	92.72	**73.28**	94.30	86.27
SC-AdaCos [26]	-	82.77	81.81	98.59	**94.01**	67.56	96.63	89.47
MFN [13]	-	83.71	90.82	98.70	91.97	71.29	96.49	88.83
STgram [20]	-	91.51	86.85	98.58	91.06	69.09	99.04	89.35
ASD-AFPA [32]	-	95.51	90.61	99.04	92.80	70.35	97.27	90.93
FTE-Net [36]	-	95.77	94.99	98.74	93.98	65.78	99.62	91.48
Unsuper-TDGCN [28]	-	-	-	-	-	-	-	-
CLP-SCF [10]	✓	95.11	91.18	98.65	93.02	69.00	99.70	91.12
AnoPatch [15]	✓	86.46	93.10	99.20	96.10	73.20	97.53	90.93
Ours	-	**98.98**	**95.02**	**99.59**	90.99	69.23	**99.81**	**92.27**

Table 5. Model Comparison on the evaluation test dataset of the DCASE 2020 Task 2 dataset. All results (%) are reported in terms of the mean of the AUC and pAUC. All the models are trained on only the normal machine sounds of the training dataset. "-" means that the result is not reported in the source paper.

DCASE 2020 Task 2								
	Pre-	Evaluation Dataset						
Models	-trained	Fan	Pump	Slider	T.Car	T.Conv	Valve	Mean
2020 No.1 [8]	-	89.42	87.69	93.68	92.04	**82.27**	93.51	89.77
SC-AdaCos [26]	-	95.42	92.53	93.54	93.96	75.00	97.31	91.30
MFN [13]	-	94.72	92.94	97.58	94.31	77.54	94.88	92.00
STgram [20]	-	-	-	-	-	-	-	-
ASD-AFPA [32]	-	-	-	-	-	-	-	-
FTE-Net [36]	-	99.72	94.78	98.17	**94.61**	69.50	93.52	91.72
Unsuper-TDGCN [28]	-	88.08	86.37	98.11	92.25	79.68	**99.87**	90.73
CLP-SCF [10]	✓	-	-	-	-	-	-	-
AnoPatch [15]	✓	95.56	94.34	99.77	96.00	83.74	96.26	94.28
Ours	-	**99.92**	**96.61**	**98.74**	94.13	70.43	93.41	**92.21**

Table 6. Model Comparison on the development test dataset of the DCASE 2023 Task 2 dataset. All results (%) are reported in terms of the Harmonic Mean (H.Mean) of the AUC and pAUC. All the models are trained on only the normal machine sounds of the training dataset. "-" means that the result is not reported in the source paper.

DCASE 2023 Task 2									
	Pre-	Development Dataset							
Models	-trained	Bearing	Fan	G.Box	Slider	T.Car	T.Train	Valve	H.Mean
2023 No.1 [4]	-	64.41	**76.27**	74.78	91.83	51.66	53.17	65.18	68.11
FeatEx [27]	-	-	-	-	-	-	-	-	66.95
MS-D2AE [3]	-	-	-	-	-	-	-	-	-
FTE-Net [36]	-	62.76	73.01	**75.97**	88.00	53.26	53.56	78.07	67.04
Han et al. [11]	✓	57.10	62.76	67.52	79.11	63.47	57.35	67.79	64.31
AnoPatch [15]	✓	70.43	66.65	58.67	81.88	58.78	67.16	53.73	64.24
Zheng et al. [38]	✓	-	-	-	-	-	-	-	65.11
Ours	-	**65.43**	67.96	71.74	**92.37**	**55.06**	**58.86**	**83.02**	**68.65**

Table 7. Model Comparison on the evaluation test dataset of the DCASE 2023 Task 2 dataset. All results (%) are reported in terms of the Harmonic Mean (H.Mean) of the AUC and pAUC. All the models are trained on only the normal machine sounds of the training dataset. "-" means that the result is not reported in the source paper.

DCASE 2023 Task 2									
	Pre-	Evaluation Dataset							
Models	-trained	B.Saw	Grinder	Shaker	T.Dro	T.Nsc	T.Tan	Vacuum	H.Mean
2023 No.1 [4]	-	60.97	65.18	63.50	55.71	84.72	60.72	92.27	66.97
FeatEx [27]	-	-	-	-	-	-	-	-	68.52
MS-D2AE [3]	-	-	-	-	-	-	-	-	66.54
FTE-Net [36]	-	**59.81**	69.69	**82.94**	57.31	**87.41**	**67.20**	88.31	**71.27**
Han et al. [11]	✓	-	-	-	-	-	-	-	-
AnoPatch [15]	✓	69.71	64.1	80.3	64.49	85.04	72.6	92.24	74.23
Zheng et al. [38]	✓	67.67	71.18	82.87	71.73	95.97	68.52	98.18	77.75
Ours	-	57.01	**72.12**	79.10	**58.43**	86.16	65.38	**88.33**	70.43

introduces domain shifts across machine types under varying operating conditions. Unlike DCASE 2021 and DCASE 2022, DCASE 2023 does not provide specific domain information; only machine attribute labels are disclosed. This setup better reflects real-world scenarios, where domains are not clearly defined, increasing the difficulty of the ASD task and emphasizing the need for DRL models to leverage additional attribute information effectively.

3.2 Evaluation Metrics

The evaluation metrics follow the official DCASE challenges [16, 6]. Three commonly used metrics are adopted for evaluating the ASD performance in this paper: area under the receiver operating characteristic curve (AUC), partial-AUC (pAUC) and the integrated scores. AUC is divided into source AUC and target AUC for the data in separate domains for DCASE 2023 challenge. pAUC is calculated as the AUC over a low false-positive-rate (FPR) range [0, 0.1]. The integrated score is the mean (for DCASE 2020) or harmonic mean (for DCASE 2023) of AUC and pAUC scores across all machine types, which is the official score used for ranking.

3.3 Implementation details

We follow the data processing methodology outlined in [36, 27]. For the DCASE 2023 Task 2 dataset, audio clips are either repeated or truncated to a fixed duration of 18 seconds, the maximum length, to handle the variability in clip lengths across machine types. For the DCASE 2020 Task 2 dataset, the audio clips are kept in their original form, each lasting 10 seconds. All audio samples are sampled at 16 kHz. Spectrograms are generated using the Short-Time Fourier Transform (STFT) with a window size of 1024 and a hop length of 512. The utterance-level spectrum is derived by applying the Fourier Transform to the entire signal. The number of classes is based on the combined categories of machine types and machine IDs (DCASE 2020) or attributes (DCASE 2023). In our experiments, T_step and F_step are configured as 32 and 8, respectively.

For training, we employ the wave-level mixup strategy [34], with the mixup coefficient drawn from Beta $\sim (0.2, 0.2)$. For non-mixup samples, label smoothing is applied with a coefficient sampled from Uniform $\sim (0, 0.5)$. We use Sub-cluster Adacos [26] as the loss function. The model is optimized using the ADAM optimizer with a learning rate of 0.001, a batch size of 64, and trained for 100 epochs on a single NVIDIA GeForce RTX 3090 GPU.

3.4 Baseline Systems

We utilize previous SOTA models as baselines. For both DCASE 2020 and DCASE 2023, we adopt the top-performing systems from the challenges [8, 4], single models employing DRL training from scratch, and models with pre-trained weights. Widely recognized methods such as Sub-cluster Adacos (SC-Adacos) [26] and MobileFaceNet (MFN) [13] are included as strong baselines.

We also integrate models specializing in analyzing machine sound spectrogram patterns [27, 32, 36, 20, 28, 3]. Additionally, we include results from pre-trained models, such as those pre-trained on large-scale speech data [11], AudioSet [15, 38], and normal data from the DCASE 2023 training set [10]. All models are trained or fine-tuned exclusively on normal machine sounds and evaluated on test datasets from both development and evaluation phases.

3.5 Comparison between Non-pre-trained Models

For the DCASE 2020 dataset, as shown in Table 4 and Table 5, our method demonstrates significant improvements over existing approaches. On the development dataset, it achieves the highest scores for the fan, pump, and toy car categories, with a mean score of 92.27, surpassing all SOTA models, including those with pre-training. On the evaluation dataset, it excels in the fan, pump, and slider categories, achieving a mean score of 92.21 and outperforming all other advanced methods without pre-training. Our approach demonstrates consistent and robust performance across various machine types and IDs, validating its efficacy in the Anomalous Sound Detection (ASD) task without domain shifts.

For the DCASE 2023 dataset, as presented in Table 6 and Table 7, the performance of all methods declines significantly compared to their results on the DCASE 2020 dataset, likely due to domain shifts caused by varying operating conditions. Despite these challenges, our model outperforms all non-pre-trained methods on most machine types. On the development dataset, our model outperform all the other methods on all machine types except for fan and gearbox, achieving the highest harmonic mean score of 68.65. On the evaluation dataset, our model's performance is comparable to the SOTA model's, and surpass the performance from the top-ranked teams in the challenge by a large margin. To obtain an overall result, we calculate the harmonic mean across all the machine types across the test of both development and evaluation datasets. As a result, our model outperforms all the other non-pre-trained methods with a harmonic mean of 69.53. These results demonstrate the robustness of our model under unknown domain variations.

3.6 Comparison with Pre-trained Models

For the DCASE 2020 dataset, as shown in Table 4 and Table 5, our method demonstrates significant improvements over existing approaches. On the development dataset, it achieves the highest scores for the fan, pump, and toy car categories, with a mean score of 92.27, surpassing all SOTA models, including those utilizing pre-training. On the evaluation dataset, our method excels in the fan, pump, and slider categories, achieving a mean score of 92.21 and outperforming all other advanced methods without pre-training. These results validate the efficacy of our approach in the Anomalous Sound Detection (ASD) task, showcasing its consistent and robust performance across various machine types and IDs, even in the absence of domain shifts.

Table 8. Results (%) on DCASE 2023 Task 2 Dataset using different scaling strategy. These values are the harmonic means of AUC and pAUC across all machine types.

Scales Strategies	Dev.	Eval.	All
no multi-scales	65.02	63.87	64.44
fix T, F multi-scale	67.29	65.33	66.30
fix F, T multi-scale	66.62	63.95	65.26
Ours	**68.65**	**70.43**	**69.53**

For the DCASE 2023 dataset, as presented in Table 6 and Table 7, the performance of all methods declines significantly compared to their results on the DCASE 2020 dataset, likely due to domain shifts caused by varying operating conditions. Despite these challenges, our model outperforms all non-pre-trained methods across most machine types. On the development dataset, it achieves the highest harmonic mean score of 68.65, outperforming all other methods on all machine types except for fan and gearbox. On the evaluation dataset, our model's performance is comparable to that of the SOTA pre-trained models and surpasses the top-ranked teams by a large margin. These results highlight the robustness of our model in handling unknown domain variations and further reinforce its effectiveness in the ASD task.

3.7 Pre-trained vs. Non-pre-trained

Table 4 to Table 7 show that adopting a pre-training and fine-tuning strategy can improve the ASD performance. However, compared to non-pre-trained models, the performance gains from pre-trained ones are not substantial. The results indicate that models designed to focus on spectrogram pattern analysis without pre-training can perform competitively well for the ASD task, even outperforming models pre-trained on speech data. This suggests that pre-training methods remain under-explored for ASD tasks. In the future, we plan to incorporate multi-scale spectrogram pattern analysis into model pre-training, which may lead to better pre-trained models for ASD tasks.

3.8 Ablation Study

Finally, we conduct ablation studies to discuss the impact of the multi-scale strategy by comparing it with a version that does not utilize the strategy. As shown in Table 8, incorporating the multi-scale scanning significantly improves the model's performance compared to the version without the multi-scale approach. This demonstrates the importance of capturing patterns across different scales for effective machine ASD. Additionally, when the scale is fixed in the frequency domain while varying scales only in the time domain, the performance is worse than when the scales are varied in the frequency domain and fixed in the time domain. This suggests that multi-scale variations in the frequency domain are more critical for learning meaningful patterns. Finally, applying multi-scale strategies to both the frequency and time domains yields the best results, which corresponds to our proposed method.

4 Conclusion

This paper introduces Multi-Scale Network (MSN), a novel DRL approach to address the challenge of extracting multi-scale spectrogram patterns for ASD. By employing kernel boxes of varying sizes and leveraging a lightweight convolutional network with shared weights, the MSN effectively captures unique characteristics of machine sounds across different scales. Experimental results on the DCASE 2020 and DCASE 2023 Task 2 datasets demonstrated that the proposed method achieves SOTA performance, highlighting its effectiveness, and potential for the ASD task.

5 Acknowledgements

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

References

1. Chen, H., Song, Y., Zhuo, Z., Zhou, Y., Li, Y.H., Xue, H., McLoughlin, I.: An effective anomalous sound detection method based on representation learning with simulated anomalies. In: Proc. of ICASSP. pp. 1–5 (2023)
2. Chen, S., Wu, Y., Wang, C., Liu, S., Tompkins, D., Chen, Z., Che, W., Yu, X., Wei, F.: BEATs: Audio pre-training with acoustic tokenizers. In: Proc. of ICML. Proceedings of Machine Learning Research, vol. 202, pp. 5178–5193. PMLR (23–29 Jul 2023)
3. Chen, S., Sun, Y., Wang, J., Wan, M., Liu, M., Li, X.: A multi-scale dual-decoder autoencoder model for domain-shift machine sound anomaly detection. Digital Signal Processing **156**, 104813 (2025)
4. Chen, S., Wang, J., Wang, J., Xu, Z.: Mdam: Multi-dimensional attention module for anomalous sound detection. In: Proc. of ICONIP. pp. 48–60 (2023)
5. Dinkel, H., Wang, Y., Yan, Z., Zhang, J., Wang, Y.: Ced: Consistent ensemble distillation for audio tagging. In: Proc. of ICASSP. pp. 291–295. IEEE (2024)
6. Dohi, K., Imoto, K., Harada, N., Niizumi, D., Koizumi, Y., Nishida, T., Purohit, H., Tanabe, R., Endo, T., Kawaguchi, Y.: Description and discussion on dcase 2023 challenge task 2: First-shot unsupervised anomalous sound detection for machine condition monitoring. In: Proc. of DCASE 2023 Workshop (2023)
7. Gemmeke, J.F., Ellis, D.P., Freedman, D., Jansen, A., Lawrence, W., Moore, R.C., Plakal, M., Ritter, M.: Audio set: An ontology and human-labeled dataset for audio events. In: Proc. of ICASSP. pp. 776–780. IEEE (2017)
8. Giri, R., Tenneti, S.V., Cheng, F., Helwani, K., Isik, U., Krishnaswamy, A.: Self-supervised classification for detecting anomalous sounds. In: Proc. of DCASE 2020 Workshop (2020)

9. Guan, J., Liu, Y., Zhu, Q., Zheng, T., Han, J., Wang, W.: Time-weighted frequency domain audio representation with gmm estimator for anomalous sound detection. In: Proc. of ICASSP. pp. 1–5 (2023)
10. Guan, J., Xiao, F., Liu, Y., Zhu, Q., Wang, W.: Anomalous sound detection using audio representation with machine id based contrastive learning pretraining. In: Proc. of ICASSP. pp. 1–5 (2023)
11. Han, B., Lv, Z., Jiang, A., Huang, W., Chen, Z., Deng, Y., Ding, J., Lu, C., Zhang, W.Q., Fan, P., et al.: Exploring large scale pre-trained models for robust machine anomalous sound detection. In: Proc. of ICASSP. pp. 1326–1330 (2024)
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proc. of CVPR. pp. 770–778 (2016)
13. Hou, Q., Jiang, A., Zhang, W.Q., Fan, P., Liu, J.: Decoupling detectors for scalable anomaly detection in aiot systems with multiple machines. In: Proc. of GLOBECOM. pp. 5937–5942 (2023)
14. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proc. of CVPR. pp. 7132–7141 (2018)
15. Jiang, A., Han, B., Lv, Z., Deng, Y., Zhang, W.Q., Chen, X., Qian, Y., Liu, J., Fan, P.: Anopatch: Towards better consistency in machine anomalous sound detection. In: Proc. of INTERSPEECH. pp. 107–111 (2024)
16. Koizumi, Y., Kawaguchi, Y., Imoto, K., Nakamura, T., Nikaido, Y., Tanabe, R., Purohit, H., Suefusa, K., Endo, T., Yasuda, M., Harada, N.: Description and discussion on dcase2020 challenge task2: Unsupervised anomalous sound detection for machine condition monitoring. In: Proc. of DCASE 2020 Workshop (2020)
17. Kolesnikov, A., Lampert, C.H.: Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In: Proc. of ECCV. pp. 695–711 (2016)
18. Kuroyanagi, I., Hayashi, T., Takeda, K., Toda, T.: Improvement of serial approach to anomalous sound detection by incorporating two binary cross-entropies for outlier exposure. In: Proc. of EUSIPCO. pp. 294–298 (2022)
19. Liu, H., Chen, Z., Yuan, Y., Mei, X., Liu, X., Mandic, D., Wang, W., Plumbley, M.D.: Audioldm: Text-to-audio generation with latent diffusion models. In: Proc. of ICML (2023)
20. Liu, Y., Guan, J., Zhu, Q., Wang, W.: Anomalous sound detection using spectral-temporal information fusion. In: Proc. of ICASSP. pp. 816–820 (2022)
21. Mai, K.T., Davies, T., Griffin, L.D., Benetos, E.: Explaining the decision of anomalous sound detectors. In: Proc. of DCASE 2022 Workshop (2022)
22. Rushe, E., Namee, B.M.: Anomaly detection in raw audio using deep autoregressive networks. In: Proc. of ICASSP. pp. 3597–3601 (2019)
23. Suefusa, K., Nishida, T., Purohit, H., Tanabe, R., Endo, T., Kawaguchi, Y.: Anomalous sound detection based on interpolation deep neural network. In: Proc. of ICASSP. pp. 271–275 (2020)
24. Tanaka, R., Tamura, S.: Few-shot anomalous sound detection based on anomaly map estimation using pseudo abnormal data. In: Proc. of ICASSP. pp. 1391–1395 (2024)
25. Vaswani, A., Shazeer, N.M., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Proc. of NIPS. vol. 30 (2017)
26. Wilkinghoff, K.: Sub-cluster adacos: Learning representations for anomalous sound detection. In: Proc. of IJCNN. pp. 1–8 (2021)
27. Wilkinghoff, K.: Self-supervised learning for anomalous sound detection. In: Proc. of ICASSP. pp. 276–280 (2024)

28. Yan, J., Cheng, Y., Wang, Q., Liu, L., Zhang, W., Jin, B.: Transformer and graph convolution-based unsupervised detection of machine anomalous sound under domain shifts. IEEE Transactions on Emerging Topics in Computational Intelligence **8**(4), 2827–2842 (2024)
29. Zavrtanik, V., Marolt, M., Kristan, M., Skočaj, D.: Anomalous sound detection by feature-level anomaly simulation. In: Proc. of ICASSP. pp. 1466–1470 (2024)
30. Zeng, X.M., Song, Y., Zhuo, Z., Zhou, Y., Li, Y.H., Xue, H., Dai, L.R., McLoughlin, I.: Joint generative-contrastive representation learning for anomalous sound detection. In: Proc. of ICASSP. pp. 1–5 (2023). https://doi.org/10.1109/ICASSP49357.2023.10095568
31. Zeng, Y., Liu, H., Xu, L., Zhou, Y., Gan, L.: Robust anomaly sound detection framework for machine condition monitoring. Tech. rep., DCASE 2022 Challenge (July 2022)
32. Zhang, H., Guan, J., Zhu, Q., Xiao, F., Liu, Y.: Anomalous Sound Detection Using Self-Attention-Based Frequency Pattern Analysis of Machine Sounds. In: Proc. of INTERSPEECH. pp. 336–340 (2023)
33. Zhang, H., Zhu, Q., Guan, J., Liu, H., Xiao, F., Tian, J., Mei, X., Liu, X., Wang, W.: First-shot unsupervised anomalous sound detection with unknown anomalies estimated by metadata-assisted audio generation. In: Proc. of ICASSP. pp. 1271–1275 (2024)
34. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization. In: Proc. of ICLR (2018)
35. Zhang, Y., Hongbin, S., Wan, Y., Li, M.: Outlier-aware Inlier Modeling and Multi-scale Scoring for Anomalous Sound Detection via Multitask Learning. In: Proc. of INTERSPEECH. pp. 5381–5385 (2023). https://doi.org/10.21437/Interspeech.2023-572
36. Zhang, Y., Liu, J., Tian, Y., Liu, H., Li, M.: A dual-path framework with frequency-and-time excited network for anomalous sound detection. In: Proc. of ICASSP. pp. 1266–1270 (2024). https://doi.org/10.1109/ICASSP48485.2024.10448126
37. Zhang, Z., Zhang, Y., Li, M.: Data augmentation by finite element analysis for enhanced machine anomalous sound detection. In: National Conference on Man-Machine Speech Communication. pp. 102–110 (2023)
38. Zheng, X., Jiang, A., Han, B., Qian, Y., Fan, P., Liu, J., Zhang, W.Q.: Improving anomalous sound detection via low-rank adaptation fine-tuning of pre-trained audio models. In: Proc. of SLT. pp. 969–974 (2024)