# FRAUNHOFER FKIE SUBMISSION FOR TASK 2: FIRST-SHOT UNSUPERVISED ANOMALOUS SOUND DETECTION FOR MACHINE CONDITION MONITORING

Technical Report

*Kevin Wilkinghoff*

Fraunhofer FKIE
Fraunhoferstraße 20, 53343 Wachtberg, Germany
kevin.wilkinghoff@fkie.fraunhofer.de

## ABSTRACT

This report contains a description of the Fraunhofer FKIE submission for task 2 “First-Shot Unsupervised Anomalous Sound Detection for Machine Condition Monitoring” of the DCASE challenge 2023. The submitted system is an adaptation of a previously proposed embedding model for extracting representations of audio data suitable for detecting anomalous sounds in domain shifted conditions. The model consists of two sub-models utilizing static and dynamic frequency information and is trained through an auxiliary classification task using the sub-cluster AdaCos loss. In this work, a modified version of mixup is presented and shown to improve the performance, especially increasing the partial area under the receiver operating characteristic curve. As a result, the proposed system is shown to significantly outperform both baseline systems of the challenge.

***Index Terms—*** anomalous sound detection, domain generalization, first-shot classification, machine listening

## 1. INTRODUCTION

Semi-supervised anomalous sound detection (ASD) for machine condition monitoring has been a task at the DCASE challenge for several years [1, 2, 3]. For all of these tasks, only normal samples have been provided as training data and the goal is to develop a system that automatically detects anomalous sounds of machines in noisy audio recordings during inference. In 2021 [2], domain shifts between a source domain with many training samples and a target domain, which consists of recordings under modified acoustic conditions caused by changed machine parameters or a different acoustic environment and for which only a few training samples are available, were introduced to the task. ASD systems needed to be adapted from source to target domains and perform well for both domains. In 2022 [3], this was extended to a domain generalization setting meaning that ASD systems need to work properly in source and target domain without needing to modify the system for particular domain shifts. This year’s challenge task is titled “First-Shot Unsupervised Anomalous Sound Detection for Machine Condition Monitoring” [4]. The main differences to previous editions of the ASD task are that 1) the development and evaluation set contain mutually exclusive machine types and 2) for each machine type only recordings of a single machine (but with different settings) are available. Thus, ASD systems submitted to the challenge cannot be fine-tuned for specific machine types on the development set and expected to perform well on the evaluation set. Furthermore, training embeddings using an auxiliary classification task is more difficult because different machine IDs cannot be used as classes. The organizers provide two baseline systems based on autoencoders using 1) the mean squared error (MSE) or 2) the Mahalanobis distance (MAHALA) as an anomaly score [5]. The dataset is a subset of ToyADMOS2 [6] and MIMII DG [7].

One particular strategy to improve the performance of ASD sytems is to simulate anomalies by modifying normal samples and teach the system to detect these simulated anomalies. However, without having access to anomalous samples it is difficult to generate realistic anomalies from scratch and therefore relatively generic methods are used for this purpose. Several works utilized data belonging to other machines of the same or other machine types [8, 9, 10] as proxy outliers. Note that using an auxiliary classification task for training the model uses the same approach implicitly. Another approach is to use data augmentation techniques and treat augmented samples as if they belong to another class (self-supervised learning). [11] used pitch shifting, time stretching and image transformations of the spectrograms and [12] used mixup [13] with a fixed small mixing coefficient. Recently, [14] proposed a method called *statistics exchange* and showed that this approach outperforms applying mixup when simulating anomalies. Statistics exchange consists of swapping first- and second-order statistics of two normal samples for randomly chosen consecutive frequency bands or time frames. A more complex procedure to simulate anomalous samples is described in [15]. Here, the authors proposed a rejection sampling algorithm that uses latent representations of an autoencoder and a Gaussian mixture model to generate anomalous sounds.

The contributions of this work are the following. First and foremost, a conceptually simple state-of-the-art first shot ASD system with strong domain generalization capabilities submitted to the DCASE challenge 2023 is presented[1]. Second, a variant of mixup for simulating anomalies during training is proposed. In experimental evaluations conducted on the development set it is shown that the proposed system significantly outperforms both baseline systems of the challenge task.

## 2. OWN BASELINE SYSTEM

The overall structure of the proposed ASD system is based on the system presented in [16] and is specifically designed for general-

[1] Open-source implementations of our baseline system and the proposed system are available at: `https://github.com/wilkinghoff/DCASE2023_task2`

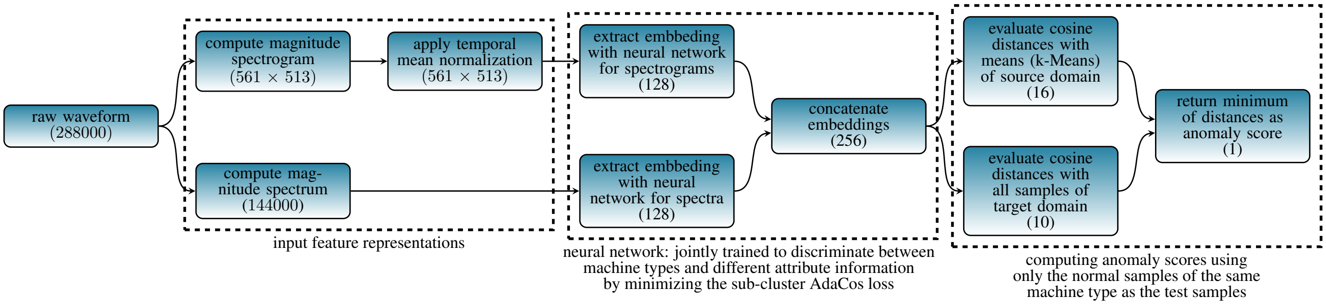
raw waveform
(288000)
compute magnitude spectrogram
(561 × 513)
apply temporal mean normalization
(561 × 513)
compute magnitude spectrum
(144000)
input feature representations
extract embedding with neural network for spectrograms
(128)
extract embedding with neural network for spectra
(128)
concatenate embeddings
(256)
neural network: jointly trained to discriminate between machine types and different attribute information by minimizing the sub-cluster AdaCos loss
evaluate cosine distances with means (k-Means) of source domain
(16)
evaluate cosine distances with all samples of target domain
(10)
return minimum of distances as anomaly score
(1)
computing anomaly scores using only the normal samples of the same machine type as the test samples

Figure 1: Structure of our own baseline system, adapted from Figure 1 in [16]. Representation size in each step is given in brackets.

izing well to multiple domains. This system is an improved version of our submissions to previous editions of the challenge task [17, 18] and is trained by solving an auxiliary classification task to learn projecting data into a suitable embedding space where, ideally, anomalous and normal samples can be easily separated. Since the proposed system closely resembles this system with only some modifications, we will now review it and use it as an additional baseline system. Note that in contrast to other state-of-the-art systems, that use different parameter settings depending on the machine type to optimize the performance [19, 20, 21] and thus are not applicable in a first-shot setting, our system uses the same parameter settings for all machine types while reaching a similar performance and thus can also be used in this year's edition of the challenge task.

Our own baseline system is depicted in Figure 1 and consists of three main blocks: 1) A frontend for computing input feature representations, 2) an embedding model for projecting the input feature representations into an embedding space and 3) a backend for computing anomaly scores. All of these three main blocks will now be discussed in more detail.

### 2.1. Frontend

To capture dynamic as well as static frequency information, two different input feature representations are used. This has been shown to significantly improve the ASD performance [16]. Before computing any feature representations, all waveforms are adjusted to have the same length by repeating all waveforms (and randomly cropping them) until they share the length of the longest waveform. As a first feature representation, we used the full magnitude spectrum to have a very high frequency resolution. Second, we used the magnitude spectrogram with a window length of 1024 and a hop size of 512 and subtracted the temporal mean to remove static frequency information.

### 2.2. Embedding model

Our baseline system utilizes the sub-cluster AdaCos loss [22], which is an angular margin loss with multiple class-centers for each class using a dynamically adaptive scale parameter as proposed in [23], This loss has been shown to outperform a standard angular margin loss when detecting anomalous sounds [22] even in domain-shifted conditions [24]. For each of the two input feature representations the model uses a specifically designed convolutional neural network (CNN) as a sub-network. The sub-network for the magnitude spectra consists of three one-dimensional convolutions and a flattening operation followed by five dense layers with 128 neurons each. For the magnitude spectrograms, a modified ResNet architecture consisting of four residual blocks, a max-pooling operation over time and a flattening operation in combination with a dense layer having 128 neurons is used. The output of both sub-networks is concatenated resulting in an embedding of size 256 for each recording. More details about the embedding model and its sub-networks can be found in [16].

The model is trained for ten epochs with a batch size of 64 by minimizing the sub-cluster AdaCos loss with 16 sub-clusters per class. For the classification task, all different machine types and values of provided attribute information are used resulting in a total of 186 classes when using all normal training samples of the development and evaluation set. To avoid learning trivial projections to the class centers, no bias terms are used and the randomly initialized class centers are not adapted during training as proposed for one-class classification in [25]. For data augmentation, mixup [13] with a mixing coefficient drawn from a uniform distribution is used.

### 2.3. Backend

The backend of the system consists of three steps. For the source domain, k-means with $k = 16$ is applied to obtain 16 means for each machine type and all cosine distances to a given test sample are computed. For the target domain, the cosine distances between all 10 normal samples belonging to a machine type and a given test sample are computed. As a last step, the anomaly score is defined as the minimum over all 26 computed cosine distances. Thus, a larger anomaly score indicates an anomalous sample while a smaller value indicates a normal sample.

## 3. PROPOSED SYSTEM

The general structure of the proposed system is the same as the baseline system presented in section 2. In this section, only the modifications of this system will be presented. As a first change, when computing the cosine distances for the source domain the means of all normal samples with the exact same attribute information have been used resulting in multiple mean embeddings for each machine type instead of applying k-Means. The other modification is a variant of mixup [13] and will now be discussed in more detail.

### 3.1. Mixup variant

To simulate anomalies during training, we propose a variant of mixup [13] as depicted in Figure 2. Classically, mixup is defined as follows: Let $x_1, x_2$ be some input data samples and $y_1, y_2 \in$

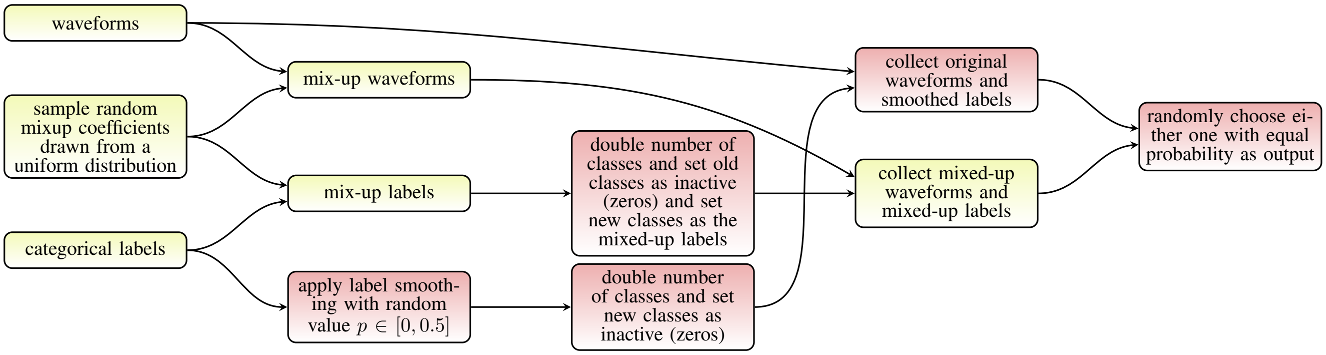
waveforms
sample random mixup coefficients drawn from a uniform distribution
categorical labels
mix-up waveforms
mix-up labels
apply label smoothing with random value $p \in [0, 0.5]$
double number of classes and set old classes as inactive (zeros) and set new classes as the mixed-up labels
double number of classes and set new classes as inactive (zeros)
collect original waveforms and smoothed labels
collect mixed-up waveforms and mixed-up labels
randomly choose either one with equal probability as output

Figure 2: Proposed mixup variant used for training the embedding model. All blocks colored in yellow are also used for standard mixup, all blocks colored in red make are only used for the proposed mixup variant.

$\{0,1\}^N$ with $\sum_{n=1}^N y_1(n) = 1 = \sum_{n=1}^N y_2(n)$ be the corresponding categorical labels where $N \in \mathbb{N}$ denotes the number of classes. Then, the mixed-up samples and labels are defined as random linear interpolations, i.e.

$$\begin{aligned} x_{\text{mixed}} &= \lambda \cdot x_1 + (1-\lambda)x_2 \\ y_{\text{mixed}} &= \lambda \cdot y_1 + (1-\lambda)y_2 \end{aligned}$$

for a random mixing coefficient $\lambda \in [0,1]$ drawn from a Beta distribution. The idea of the proposed mixup variant is to use randomly mixed-up samples as simulated anomalous samples similar to the idea proposed in [12]. But instead of only using mixed-up samples as an additional anomalous class with a fixed (small) mixing coefficient, the model needs to recognize the classes that are mixed-up as well as the mixing coefficient as it is the case for standard mixup, too. This can also be seen as some type of self-supervised learning. In the proposed variant, mixup is applied by doubling the number of classes and using one half of the classes for original data samples that have not been mixed, and using the other half for mixed-up data. Hence, the original samples as well as the mixed-up samples have to be recognized correctly but the model needs to be able to discriminate between original and mixed-up samples. Furthermore, label smoothing [26] with a random value $p \in [0, \frac{1}{2}]$ is applied to the original labels. This means that a categorical label $y \in \{0,1\}^N$ with $\sum_{n=1}^N y(n) = 1$ is replaced with

$$y_{\text{smoothed},p} = \begin{cases} 1 - p + \frac{p}{N} & \text{if } y(n) = 1 \\ \frac{p}{N} & \text{if } y(n) = 0. \end{cases}$$

Here, the idea of applying label smoothing is to avoid that the model overfits to the classes and also to relax the requirements of strictly differentiating between non-mixed and mixed data samples. During training, the mixed-up waveforms and labels, or the original waveforms and smoothed labels are randomly chosen to be provided as training data. Note that in contrast to many other systems, the waveforms and not the spectral representations of the data are mixed for our proposed approach. This has the advantage that both input feature representations utilize the same (or no) mixing coefficient.

### 3.2. Setting a decision threshold

In a semi-superved ASD setting, a decision threshold has to be estimated using normal samples only. This is a very difficult task by itself. To our best knowledge, the only viable strategy is to find a decision threshold that separates the extreme values of the anomaly scores belonging to the normal samples from the rest and hope that this threshold also works well for separating anomaly scores belonging to anomalous samples from those belonging to normal samples [27]. There are several methods available for estimating a decision threshold. In [27], it has been shown that multi-stage methods outperform single-stage methods but in most cases the differences in performance between different methods are only marginal. Since the final metric of the challenge task does not include a decision threshold and thus estimating a good decision threshold is not important for the challenge, we simply assumed a uniform distribution of the anomaly scores and used the 90th percentile as the decision threshold as done in previous years [17, 18].

## 4. SUBMISSIONS

In total, three different systems have been submitted to the challenge. The first submission is an ensemble consisting of the mean of the anomaly scores obtained with ten independently trained versions of our baseline system as presented in [16]. The second submission is an ensemble consisting of the maximum of the anomaly scores belonging to ten independently trained versions of the proposed systems. As a third submission, we submitted an ensemble of the other two submissions by taking the maximum of the anomaly scores belonging to both systems. Hence, this ensemble consists of twenty subsystems.

## 5. RESULTS

The experimental results obtained on the development set with our three submitted systems as well as both baseline systems of the challenge can be found in Table 1. On the target domain, it can be seen that the presented systems, in general, perform much better on than both baseline systems. Although on the source domain the performance of our systems is worse in some cases, namely for the machine types “ToyCar” and “ToyConveyor”, the performance of our systems on mixed domains is much better or at least comparable. Overall, the performance of our presented systems is significantly better than the ones obtained with the baseline systems.

When comparing the performance of our own baseline system to the proposed system, for some machine types as for example “ToyCar” and “fan” the performance degrades with the proposed changes but for others, most notably the machine type “valve” the

Table 1: AUCs and pAUCs per machine type obtained on the development set with both baseline systems of the challenge, the ASD system presented in [16] and the proposed system as well as an ensemble of both systems. The last row contains the harmonic mean taken over all machine types. Highest AUCs and pAUCs in each row are highlighted in bold letters.

| dataset split | | baseline systems | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | MSE [5] | | MAHALA [5] | | own baseline [16] | | proposed system | | ensemble | |
| machine type | domain | AUC | pAUC | AUC | pAUC | AUC | pAUC | AUC | pAUC | AUC | pAUC |
| ToyCar | source | 69.96% | **49.05%** | **76.44%** | 48.00% | 53.32% | 48.00% | 50.04% | 48.00% | 52.36% | 48.00% |
| ToyCar | target | 47.28% | **57.47%** | 45.56% | 49.68% | **68.00%** | 53.47% | 63.36% | 51.79% | 63.68% | 51.79% |
| ToyCar | mixed | 58.62% | **53.63%** | **61.00%** | 49.58% | 60.32% | 48.00% | 55.98% | 48.79% | 57.04% | 48.79% |
| ToyTrain | source | **57.86%** | 47.58% | 54.96% | 47.37% | 48.52% | 47.79% | 55.60% | **48.22**% | 55.60% | 48.21% |
| ToyTrain | target | 57.18% | 49.68% | 41.64% | 49.47% | **67.72%** | **54.53%** | 63.64% | 53.05% | 63.72% | 53.05% |
| ToyTrain | mixed | 57.52% | 48.32% | 48.30% | 48.00% | 57.92% | 48.37% | 59.71% | **50.11%** | **59.76%** | **50.11%** |
| bearing | source | 65.24% | 59.58% | 64.62% | 60.42% | 82.40% | 64.00% | **85.00%** | **76.63%** | **85.00%** | **76.63%** |
| bearing | target | 54.74% | 48.00% | 53.02% | **49.47%** | 68.56% | 47.58% | **71.64%** | 48.63% | **71.64%** | 48.63% |
| bearing | mixed | 59.99% | 49.89% | 58.82% | 50.05% | 75.57% | 51.42% | **78.16%** | **56.47%** | **78.16%** | **56.47%** |
| fan | source | 73.16% | 56.21% | 79.12% | 56.21% | **86.80%** | **66.95%** | 82.76% | 61.68% | 82.84% | 61.68% |
| fan | target | 31.98% | 63.37% | 36.64% | **63.79%** | **73.64%** | 51.37% | 65.76% | 53.68% | 66.12% | 53.68% |
| fan | mixed | 52.57% | **59.37%** | 57.88% | 59.26% | **78.96%** | 52.32% | 72.23% | 53.00% | 72.46% | 53.00% |
| gearbox | source | 60.18% | 52.00% | 71.82% | 57.68% | 84.48% | **68.84%** | **87.60%** | 66.53% | **87.60%** | 66.53% |
| gearbox | target | 60.26% | 56.00% | 70.50% | 55.79% | 80.84% | 60.63% | **84.92%** | **62.53%** | 84.68% | **62.53%** |
| gearbox | mixed | 60.22% | 53.79% | 71.16% | 56.37% | 82.38% | **65.21%** | **85.55%** | 63.26% | 85.41% | 63.26% |
| slide rail | source | 69.54% | 59.58% | 83.96% | 61.05% | 99.36% | 96.63% | **99.60%** | **97.89%** | **99.60%** | **97.89%** |
| slide rail | target | 47.30% | 52.00% | 74.28% | 49.89% | 88.68% | 68.63% | **93.72%** | **77.05%** | **93.72%** | **77.05%** |
| slide rail | mixed | 58.42% | 56.63% | 79.12% | 54.21% | 91.96% | 72.68% | **95.71%** | **81.95%** | 95.64% | **81.95%** |
| valve | source | 56.90% | 54.95% | 55.26% | 52.63% | 91.60% | 64.63% | **99.24%** | **96.21%** | 98.88% | 94.32% |
| valve | target | 51.52% | 50.74% | 51.96% | 50.74% | 86.36% | 58.53% | **98.96%** | **94.53%** | **98.96%** | **94.53%** |
| valve | mixed | 54.21% | 51.10% | 53.61% | 50.84% | 87.02% | 59.95% | **98.54%** | **92.63%** | 98.13% | 91.32% |
| all | mixed | 57.23% | 53.01% | 59.97% | 52.35% | 74.27% | 55.62% | 74.84% | **60.43%** | **75.10%** | 60.35% |

performance improves. Overall, the AUC score of both systems is very similar but the pAUC score of the proposed system is significantly higher. Hence, the proposed changes appear to improve overall performance. Using an ensemble of both systems leads to a very similar performance as the proposed system alone and thus training such a large ensemble does not seem to be helpful and certainly not to be necessary. The final results of the challenge are expected to give more insights on this by measuring how effective the proposed changes really are.

## 6. CONCLUSIONS

In this work, a first-shot ASD system for task 2 of the DCASE challenge 2023 has been presented. The system is based on a previously proposed embedding model trained by using an auxiliary classification task and a novel mixup variant for ASD. In experiments conducted on the development set of the challenge task, it has been shown that the proposed system significantly outperforms both baseline systems of the challenge. Furthermore, training the embedding model with the proposed mixup variant helps to improve the performance, especially by increasing the resulting pAUC. For future work, it is planned to conduct additional ablation studies and compare the performance of our proposed system to the ones obtained with other systems submitted to task 2 of the DCASE challenge 2023.

## 7. REFERENCES

[1] Y. Koizumi, *et al.*, "Description and discussion on DCASE2020 challenge task2: Unsupervised anomalous sound detection for machine condition monitoring," in *5th Workshop on Detection and Classification of Acoustic Scenes and Events*, 2020, pp. 81–85.

[2] Y. Kawaguchi, *et al.*, "Description and discussion on dcase 2021 challenge task 2: Unsupervised anomalous detection for machine condition monitoring under domain shifted conditions," in *6th Workshop on Detection and Classification of Acoustic Scenes and Events*, 2021, pp. 186–190.

[3] K. Dohi, *et al.*, "Description and discussion on DCASE 2022 challenge task 2: Unsupervised anomalous sound detection for machine condition monitoring applying domain generalization techniques," in *7th Workshop on Detection and Classification of Acoustic Scenes and Events*. Tampere University, 2022, pp. 26–30.

[4] ——, "Description and discussion on DCASE 2023 challenge task 2: First-shot unsupervised anomalous sound detection for machine condition monitoring," 2023, arXiv:2305.07828.

[5] N. Harada, D. Niizumi, D. Takeuchi, Y. Ohishi, and M. Yasuda, "First-shot anomaly detection for machine

condition monitoring: A domain generalization baseline," *arXiv:2303.00455*, 2023.

[6] 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 *6th Workshop on Detection and Classification of Acoustic Scenes and Events*, 2021, pp. 1–5.

[7] K. Dohi, *et al.*, "MIMII DG: sound dataset for malfunctioning industrial machine investigation and inspection for domain generalization task," in *7th Workshop on Detection and Classification of Acoustic Scenes and Events*. Tampere University, 2022, pp. 26–30.

[8] P. Primus, V. Haunschmid, P. Praher, and G. Widmer, "Anomalous sound detection as a simple binary classification problem with careful selection of proxy outlier examples," in *5th Workshop on Detection and Classification of Acoustic Scenes and Events*, 2020, pp. 170–174.

[9] I. Kuroyanagi, T. Hayashi, Y. Adachi, T. Yoshimura, K. Takeda, and T. Toda, "An ensemble approach to anomalous sound detection based on conformer-based autoencoder and binary classifier incorporated with metric learning," in *6th Workshop on Detection and Classification of Acoustic Scenes and Events*, 2021, pp. 110–114.

[10] K. Dohi, T. Endo, H. Purohit, R. Tanabe, and Y. Kawaguchi, "Flow-based self-supervised density estimation for anomalous sound detection," in *International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2021, pp. 336–340.

[11] T. Inoue, *et al.*, "Detection of anomalous sounds for machine condition monitoring using classification confidence," in *Detection and Classification of Acoustic Scenes and Events Workshop*, 2020, pp. 66–70.

[12] J. A. Lopez, H. Lu, P. Lopez-Meyer, L. Nachman, G. Stemmer, and J. Huang, "A speaker recognition approach to anomaly detection," in *Detection and Classification of Acoustic Scenes and Events Workshop*, 2020, pp. 96–99.

[13] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," in *International Conference on Learning Representations*, 2018.

[14] H. Chen, *et al.*, "An effective anomalous sound detection method based on representation learning with simulated anomalies," in *International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2023.

[15] Y. Koizumi, S. Saito, H. Uematsu, Y. Kawachi, and N. Harada, "Unsupervised detection of anomalous sound based on deep learning and the neyman-pearson lemma," *IEEE ACM Trans. Audio Speech Lang. Process.*, vol. 27, no. 1, pp. 212–224, 2019.

[16] K. Wilkinghoff, "Design choices for learning embeddings from auxiliary tasks for domain generalization in anomalous sound detection," in *International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2023.

[17] ——, "Utilizing sub-cluster AdaCos for anomalous sound detection under domain shifted conditions," DCASE Challenge, Tech. Rep., 2021.

[18] ——, "An outlier exposed anomalous sound detection system for domain generalization in machine condition monitoring," DCASE Challenge, Tech. Rep., 2022.

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

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

[21] F. Xiao *et al.*, "The DCASE2022 challenge task 2 system: Anomalous sound detection with self-supervised attribute classification and GMM-based clustering," DCASE Challenge, Tech. Rep., 2022.

[22] K. Wilkinghoff, "Sub-cluster AdaCos: Learning representations for anomalous sound detection," in *International Joint Conference on Neural Networks*. IEEE, 2021.

[23] X. Zhang, R. Zhao, Y. Qiao, X. Wang, and H. Li, "AdaCos: Adaptively scaling cosine logits for effectively learning deep face representations," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2019, pp. 10 823–10 832.

[24] K. Wilkinghoff, "Combining multiple distributions based on sub-cluster AdaCos for anomalous sound detection under domain shifted conditions," in *Detection and Classification of Acoustic Scenes and Events Workshop*, 2021, pp. 55–59.

[25] L. Ruff, *et al.*, "Deep one-class classification," in *35th International Conference on Machine Learning*, vol. 80. PMLR, 2018, pp. 4390–4399.

[26] R. Müller, S. Kornblith, and G. E. Hinton, "When does label smoothing help?" in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems*, 2019, pp. 4696–4705.

[27] K. Wilkinghoff and A. Cornaggia-Urrigshardt, "On choosing decision thresholds for anomalous sound detection in machine condition monitoring," in *24th International Congress on Acoustics*. The Acoustical Society of Korea, 2022.