

ADAPROJ: ADAPTIVELY SCALED ANGULAR MARGIN SUBSPACE PROJECTIONS FOR ANOMALOUS SOUND DETECTION WITH AUXILIARY CLASSIFICATION TASKS

Kevin Wilkinghoff

Fraunhofer FKIE, Fraunhoferstr. 20 53353 Wachtberg, Germany
kevin.wilkinghoff@ieee.org

ABSTRACT

The state-of-the-art approach for semi-supervised anomalous sound detection is to first learn an embedding space by using auxiliary classification tasks based on meta information or self-supervised learning and then estimate the distribution of normal data. In this work, AdaProj a novel loss function for training the embedding model is presented. In contrast to commonly used angular margin losses, which project data of each class as close as possible to their corresponding class centers, AdaProj learns to project data onto class-specific subspaces while still ensuring an angular margin between classes. By doing so, the resulting distributions of the embeddings belonging to normal data are not required to be as restrictive as other loss functions allowing a more detailed view on the data. In experiments conducted on the DCASE2022 and DCASE2023 anomalous sound detection datasets, it is shown that using AdaProj to learn an embedding space significantly outperforms other commonly used loss functions.

Index Terms— machine listening, anomaly detection, representation learning, domain generalization

1. INTRODUCTION

Semi-supervised anomaly detection is the task of training a system to differentiate between normal and anomalous data using only normal training samples [1]. An example application is acoustic machine condition monitoring for predictive maintenance [2, 3]. Here, normal data corresponds to sounds of fully functioning machines whereas anomalous sounds indicate mechanical failure. One of the main difficulties to overcome in acoustic machine condition monitoring is that it is practically impossible to record isolated sounds of a target machine. Instead, recordings also contain many other sounds emitted by non-target machines or other sound sources such as humans. Compared to this complex acoustic scene, anomalous signal components of the target machines are very subtle and hard to detect without utilizing additional knowledge. Another main difficulty is that a system should also be able to reliably detect anomalous sounds when changing the acoustic conditions or machine settings without needing to collect large amounts of data in these changed conditions or to re-train the system (domain generalization [4]). One possibility to overcome both difficulties is to learn a mapping of the audio signals into a fixed-dimensional vector space, in which representations belonging to normal and anomalous data, called embeddings, can be easily separated. Then, by estimating the distribution of normal training samples in the embedding space, one can compute an anomaly score for a test sample to distinguish between normal and anomalous samples.

To train such an embedding model, the state-of-the-art is to utilize an auxiliary classification task using provided meta information

or self-supervised learning. This enables the embedding model to closely monitor target signals and ignore other signals and noise [5]. For machine condition monitoring, possible auxiliary tasks are classifying between machine types [6–8] or, additionally, between different machine states and noise settings [9–11], recognizing augmented and non-augmented versions of normal data [6, 12] or predicting the activity of machines [10]. Using an auxiliary task to learn embeddings is also called outlier exposure (OE) [13] because normal samples belonging to other classes than a target class can be considered proxy outliers [14].

The contributions of this work are the following. First and foremost, AdaProj, a novel angular margin loss function that learns class-specific subspaces for training an embedding model, is presented¹. Second, it is proven that AdaProj has arbitrary large optimal solution spaces allowing to relax the compactness requirements of the class-specific distributions in the embedding space. Last but not least, AdaProj is compared to other commonly used loss functions. In experiments conducted on the DCASE2022 and DCASE2023 anomalous sound detection (ASD) datasets it is shown that AdaProj outperforms other commonly used loss functions.

1.1. Related Work

When training a neural network to solve a classification task, usually the softmax function in combination with the categorical cross-entropy (CXE) is used. However, this only reduces inter-class similarity without explicitly increasing intra-class similarity [15]. When training an embedding model for anomaly detection, high intra-class similarity is a desired property to cluster normal data and be able to detect anomalous samples. There are several loss functions that explicitly increase intra-class similarity: [16] proposed a compactness loss to project the data into a hypersphere of minimal volume for one-class classification. However, for machine condition monitoring in noisy conditions it is known that one-class losses perform worse than losses that also discriminatively solve an auxiliary classification task [5]. [17] utilized an additional descriptiveness loss consisting of a CXE imposing a classification task on another arbitrary dataset than the target dataset to regularize the training objective. For machine condition monitoring, often meta information is available as it can at least be ensured which machine is being recorded when collecting data. [8] used center loss [18], which minimizes the distance to learned class centers for each class. Another choice are angular margin losses that learn an embedding space on the unit sphere while ensuring a margin between classes, which improves the generalization capabilities. Specific examples are the additive margin softmax loss [15] as used by [7, 19] and ArcFace [20]

¹An open-source implementation of the AdaProj loss is available at: <https://github.com/wilkinghoff/AdaProj>

as used by [6, 11, 21]. [22, 23] use the AdaCos loss [24], which essentially is ArcFace with an adaptive scale parameter, or the sub-cluster AdaCos loss [25], which utilizes multiple sub-clusters for each class instead of a single one.

As stated before, the goal of this work is to reduce the restrictions on the learned distributions in the embedding space by learning class-specific linear subspaces. There are also other works on losses aiming at learning subspaces based on orthogonal projections in an embedding space. [26] used orthogonal projections as a constraint for training an autoencoder based anomaly detection system. Another example is semi-supervised image classification by using a combination of class-specific subspace projections with a reconstruction loss and ensure that they are different by also using a discriminative loss [27]. Our work focuses on learning an embedding space through an auxiliary classification task that is well-suited for semi-supervised anomaly detection.

2. METHODOLOGY

2.1. Notation

Let $\phi : X \rightarrow \mathbb{R}^D$ denote a neural network where X denotes some input space, which consists of audio signals in this work, and $D \in \mathbb{N}$ denotes the dimension of the embedding space. Define the linear projection of $x \in \mathbb{R}^D$ onto the subspace $\text{span}(\mathcal{C}_k) \subset \mathbb{R}^D$ as $P_{\text{span}(\mathcal{C}_k)}(x) := \sum_{c_k \in \mathcal{C}_k} \langle x, c_k \rangle c_k$. Furthermore, let $\mathcal{S}^{D-1} = \{y \in \mathbb{R}^D : \|y\|_2 = 1\} \subset \mathbb{R}^D$ denote the D -sphere and define $P_{\mathcal{S}^{D-1}}(x) := \frac{x}{\|x\|_2} \in \mathcal{S}^{D-1}$ to be the projection onto the D -sphere.

2.2. AdaProj loss function

Similar to the sub-cluster AdaCos loss [25], the idea of the AdaProj loss is to enlarge the space of optimal solutions to allow the network to learn less restrictive distributions of normal data. This relaxation is achieved by measuring the distance to class-specific subspaces while training the embedding model instead of measuring the distance to a single or multiple centers as done for other angular margin losses and may help to differentiate between normal and anomalous embeddings after training. The reason is that for some auxiliary classes a strong compactness may be detrimental when aiming to learn an embedding space that separates normal and anomalous data since both may be distributed very similarly.

Formally, the definition of the AdaProj loss is as follows.

Definition 1 (AdaProj loss). Let $\mathcal{C}_k \subset \mathbb{R}^D$ with $|\mathcal{C}_k| = J \in \mathbb{N}$ denote class centers for class $k \in \{1, \dots, N_{\text{classes}}\}$. Then for the AdaProj loss the logit for class $k \in \{1, \dots, N_{\text{classes}}\}$ is defined as

$$L(x, \mathcal{C}_k) := \hat{s} \cdot \|P_{\mathcal{S}^{D-1}}(x) - P_{\mathcal{S}^{D-1}}(P_{\text{span}(\mathcal{C}_k)}(x))\|_2^2$$

where $\hat{s} \in \mathbb{R}_+$ is the adaptive scale parameter of the AdaCos loss [24]. Inserting these logits into a softmax function and computing the CXE yields the AdaProj loss function.

Remark. Note that, by Lemma 5 of [5], it holds that

$$\begin{aligned} & \|P_{\mathcal{S}^{D-1}}(x) - P_{\mathcal{S}^{D-1}}(P_{\text{span}(\mathcal{C}_k)}(x))\|_2^2 \\ &= 2(1 - \langle P_{\mathcal{S}^{D-1}}(x), P_{\mathcal{S}^{D-1}}(P_{\text{span}(\mathcal{C}_k)}(x)) \rangle), \end{aligned}$$

which is equal to the cosine distance in this case and explains why the AdaProj loss can be called an angular margin loss.

As for other angular margin losses, projecting the embedding space onto the D -sphere has several advantages [5]. Most importantly, if D is sufficiently large randomly initialized centers are with very high probability approximately orthonormal to each other [28], i.e. distributed equidistantly, and sufficiently far away from $\mathbf{0} \in \mathbb{R}^D$. Therefore, one does not need to carefully design a method to initialize the centers. Another advantage is that a normalization may prevent numerical issues, similar to applying batch normalization [29].

The following Lemma shows that using the AdaProj loss, as defined above, indeed increases the solution space.

Lemma 2. Let $x \in \mathbb{R}^D$ and let $\mathcal{C} \subset \mathbb{R}^D$ contain pairwise orthonormal elements. If $x \in \text{span}(\mathcal{C}) \cap \mathcal{S}^{D-1}$, then

$$\|P_{\mathcal{S}^{D-1}}(x) - P_{\mathcal{S}^{D-1}}(P_{\text{span}(\mathcal{C})}(x))\|_2^2 = 0.$$

Proof. Let $x \in \text{span}(\mathcal{C}) \cap \mathcal{S}^{D-1} \subset \mathbb{R}^D$ with $|\mathcal{C}| = J$. Therefore, $\|x\|_2 = 1$ and there are $\lambda_j \in \mathbb{R}$ with $x = \sum_{j=1}^J \lambda_j c_j$. Thus, it holds that

$$\begin{aligned} x &= \sum_{j=1}^J \lambda_j c_j = \sum_{j=1}^J \sum_{i=1}^J \lambda_i \langle c_i, c_j \rangle c_j = \sum_{j=1}^J \left(\sum_{i=1}^J \lambda_i \langle c_i, c_j \rangle \right) c_j \\ &= \sum_{j=1}^J \langle x, c_j \rangle c_j = P_{\text{span}(\mathcal{C})}(x). \end{aligned}$$

Hence, we obtain

$$\|P_{\mathcal{S}^{D-1}}(x) - P_{\mathcal{S}^{D-1}}(P_{\text{span}(\mathcal{C})}(x))\|_2^2 = 0. \quad \square$$

Remark. If \mathcal{C} contains randomly initialized elements of the unit sphere and D is sufficiently large, then the elements of \mathcal{C} are approximately pairwise orthonormal with very high probability [28].

When inserting the projection onto the $D - 1$ -sphere as an operation into the neural network, this Lemma shows that the solution space for the AdaProj loss function is increased to the whole subspace $\text{span}(\mathcal{C})$, which has a dimension of $|\mathcal{C}|$ with very high probability. Because of this, it should be ensured that $|\mathcal{C}| < D$. Otherwise the whole embedding space may be an optimal solution and thus the network cannot learn a meaningful embedding space. In comparison, for the AdaCos loss only the class centers themselves are optimal solutions and for the sub-cluster AdaCos loss each sub-cluster is an optimal solution [5].

3. EXPERIMENTAL RESULTS

3.1. Datasets and performance metrics

For the experiments, the DCASE2022 ASD dataset [2] and the DCASE2023 ASD dataset [3] for semi-supervised machine condition monitoring were used. Both datasets consist of a development set and an evaluation set that are divided into a training split containing only normal data and a test split containing normal as well as anomalous data. Furthermore, both tasks explicitly capture the problem of domain generalization [4] by defining a source and a target domain, which differs from the source domain by altering machine parameters or noise conditions. The task is to detect anomalous samples regardless of the domain a sample belongs to by training a system with only normal data. As meta information, the

target machine type of each sample is known and for the training samples, also the domain and additional parameter settings or noise conditions, called attribute information, are known and thus can be utilized to train an embedding model.

The DCASE2022 ASD dataset [2] consists of the machine types “ToyCar” and “ToyTrain” from ToyAdmos2 [30] and “fan”, “gearbox”, “bearing”, “slide rail” and “valve” from MIMII-DG [31]. For each machine type, there are six different sections corresponding to different domain shifts and also defining subsets used for computing the performance. These sections are known for each recording and can also be utilized as meta information to train the system. For the source domain of each section, there are 1000 normal audio recordings with a duration of 10 s and a sampling rate of 16 kHz belonging to the training split. For the target domain of each section, there are only 10 normal audio recordings belonging to the training split. The test splits of each section contain approximately 100 normal and 100 anomalous samples.

The DCASE2023 ASD dataset [3] is similar to the DCASE2022 ASD dataset with the following modifications. First of all, the development set and the evaluation set contain mutually exclusive machine types. More concretely, the development set contains the same machine types as the DCASE2022 dataset and the evaluation set contains the machine types “ToyTank”, “ToyNscale” and “ToyDrone” from ToyAdmos2+ [32] and “vacuum”, “band-saw”, “grinder” and “shaker” from MIMII-DG [31]. Furthermore, there is only a single section for each machine type, which makes the auxiliary classification task much easier resulting in less informative embeddings for the ASD task. Last but not least, the duration of each recording has a length between 6 s and 18 s. Overall, all three modifications make this task much more challenging than the DCASE2022 ASD task.

To measure the performance of the ASD systems the threshold-independent area under the receiver operating characteristic (ROC) curve (AUC) metric is used. In addition, the partial area under the ROC curve (pAUC) [33], which is the AUC for low false positive rates ranging from 0 to p , with $p = 0.1$, is used. Both performance metrics are computed domain-independent for every previously defined section of the dataset and the harmonic mean of all resulting values is the final score used to measure and compare the performances of different ASD systems.

3.2. Anomalous sound detection system

For all experiments conducted in this work, the state-of-the-art ASD system presented in [23] is used. An overview of the system can be found in Figure 1. The system consists of three main components: 1) a feature extractor, 2) an embedding model and 3) a backend for computing anomaly scores.

In the first processing block, two different feature representations are extracted from the raw waveforms, namely magnitude spectrograms and the magnitude spectrum. To capture less similar information with both feature representations, the temporal mean is subtracted from the spectrograms, essentially removing static frequency information that are captured with the highest possible resolution by the spectra.

For each of the two feature representations, another convolutional sub-network is trained and the resulting embeddings are concatenated and normalized with respect to the Euclidean norm to obtain a single embedding. In contrast to the original architecture, the embedding dimension is doubled from 256 to 512 to increase the likelihood of two randomly initialized center vectors to be orthogo-

nal. Note that even if some of the randomly sampled class centers are not orthogonal, the probability that they are linearly independent is equal to 1 if $J < D$. Thus, the subspaces spanned by the class centers do not collapse. More details about the subnetwork architectures can be found in [23]. The network is trained for 10 epochs using a batch size of 64 using adam [34] by utilizing meta information such as machine types and the provided attribute information as an auxiliary classification task. Different loss functions can be used for this purpose and will be compared in the next subsection. All loss functions investigated in this work require class-specific center vectors, which are initialized randomly using Glorot uniform initialization [35]. To improve the ASD performance, the class centers are not adapted during training and no bias terms are used as proposed in [16] for one-class classification. Furthermore, mixup [36] with a uniformly distributed mixing coefficient is applied to the waveforms.

As a backend, k-means with 32 means is applied to the normal training samples of the source domain. For a given test sample, the smallest cosine distance to these means and the ten normal training samples of the target domain is used as an anomaly score. Thus, smaller values indicate normal samples whereas higher values indicate anomalous samples.

3.3. Performance evaluation

In the first experiment, the ASD performance obtained with the following loss functions was compared: 1) individual class-specific intra-class (IC) compactness losses jointly trained on all classes [16] 2) an additional discriminative CXE loss, similar to the descriptiveness loss used in [17] but trained on the same dataset, 3) the AdaCos loss [24], 4) the sub-cluster AdaCos loss [25] with 32 sub-clusters and 5) the proposed AdaProj loss. Each experiment was repeated ten times to reduce the variance of the resulting performances. The results can be found in Table 1.

The main observation to be made is that the proposed AdaProj loss outperforms all other losses. Especially on the DCASE2023 dataset, there are significant improvements to be observed. The most likely explanation is that for this dataset the classification task is less difficult and thus a few classes may be easily identified leading to embeddings that do not carry enough information to distinguish between embeddings belonging to normal and anomalous samples of these classes.

Another interesting observation is that, in contrast to the original results presented in [25], the sub-cluster AdaCos loss actually performs slightly worse than the AdaCos loss despite having a higher solution space. A possible explanation is that in [25], the centers are adapted during training whereas, in our work, they are not as this has been shown to improve the resulting performance [23]. Since all centers have approximately the same distance to each other when being randomly initialized, i.e. the centers belonging to a target class and the other centers, the network will likely utilize only a single center for each class that is closest to the initial embeddings of the corresponding target class. Moreover, a low inter-class similarity is more difficult to ensure due to the higher total number of sub-clusters belonging to other classes. This leads to more restrictive requirements when learning class-specific distributions and thus actually reduces the ability to differentiate between embeddings belonging to normal and anomalous samples.

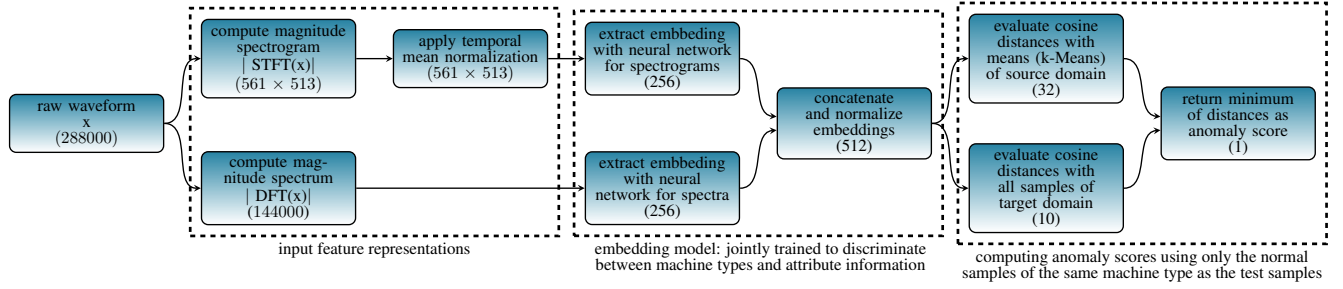


Figure 1: Structure of the ASD system, adapted from Figure 1 in [23]. Representation size in each step is given in brackets.

Table 1: ASD performance obtained with different loss functions. Harmonic means of all AUCs and pAUCs over all pre-defined sections of the dataset are depicted in percent. Arithmetic mean and standard deviation of the results over ten independent trials are shown. Best results in each column are highlighted with bold letters.

loss function	DCASE2022 dev. set [2]		DCASE2022 eval. set [2]		DCASE2023 dev. set [3]		DCASE2023 eval. set [3]		arithmetic mean	
	AUC	pAUC	AUC	pAUC	AUC	pAUC	AUC	pAUC	AUC	pAUC
IC compactness loss [16]	79.2 ± 0.9	64.7 ± 1.1	70.3 ± 0.8	58.9 ± 0.8	67.7 ± 1.2	56.9 ± 0.9	64.0 ± 1.5	55.8 ± 0.9	70.3	59.1
IC compactness loss + CXE [17]	79.0 ± 0.8	65.0 ± 0.7	72.6 ± 0.4	60.3 ± 0.7	70.4 ± 1.0	57.4 ± 1.1	67.5 ± 0.8	57.5 ± 1.0	72.4	60.1
AdaCos loss [24]	79.8 ± 0.7	65.5 ± 0.9	73.0 ± 0.4	59.7 ± 0.6	70.9 ± 0.9	56.8 ± 0.9	68.0 ± 1.6	58.0 ± 1.1	72.9	60.0
sub-cluster AdaCos loss [25]	80.0 ± 1.4	65.2 ± 1.1	72.9 ± 0.6	59.5 ± 0.5	70.4 ± 0.9	56.3 ± 0.8	66.5 ± 1.6	56.2 ± 1.0	72.5	59.3
proposed AdaProj loss	80.6 ± 0.8	65.5 ± 1.3	73.6 ± 0.7	60.5 ± 0.7	71.4 ± 1.0	56.2 ± 0.7	69.8 ± 1.3	60.0 ± 0.5	73.9	60.6

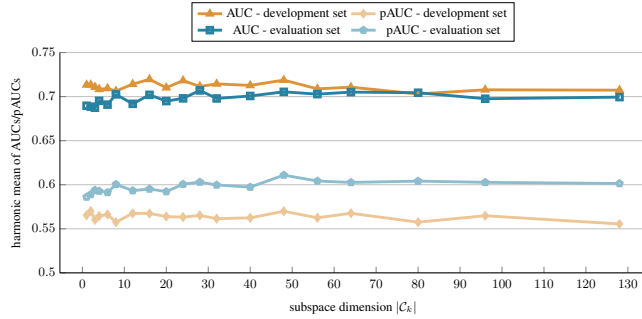


Figure 2: Domain-independent performance obtained on the DCASE2023 dataset with different subspace dimensions. The means over ten independent trials are shown.

3.4. Investigating the impact of the subspace dimension on the performance

As an ablation study, different choices for the dimension of the subspaces have been compared experimentally on the DCASE2023 ASD dataset. The results can be found in Figure 2. It can be seen, that, on the development set, the results are relatively stable while a larger dimension slightly improves the performance on the evaluation set without any significant differences. For subspace dimensions greater than 48 the performances seem to slightly degrade again. In conclusion, the subspace dimension should be neither too high nor too low but other than that appears to not have a significant impact on the performance. Thus, a dimension of 32, as used for the other experiments in this work, appears to be a reasonable choice. Since using the AdaProj with this subspace dimension also outperformed the other loss functions on the DCASE2022 ASD dataset (cf. Table 1, this particular dimension may serve as a default hyperparameter setting for the AdaProj loss.

4. CONCLUSIONS

In this work, AdaProj a novel angular margin loss function specifically designed for semi-supervised anomaly detection with auxiliary classification tasks was presented. It was proven that this loss function learns an embedding space with class-specific subspaces of arbitrary dimension. In contrast to other angular margin losses, which try to project data to individual points in space, this relaxes the requirements of solving the classification task and allows for less compact distributions in the embedding space. In experiments conducted on the DCASE2022 and DCASE2023 ASD datasets, it was shown that using AdaProj results in better performance than other commonly used loss functions. In conclusion, the resulting embedding space has a more desirable structure than the other embedding spaces for differentiating between normal and anomalous samples. For future work, it is planned to evaluate AdaProj on other datasets and further improve the performance of the ASD system by utilizing self-supervised learning [37] or multi-task learning [9]. Investigating how the AdaProj loss performs for supervised or unsupervised tasks in comparison to other loss functions may also be of interest.

5. REFERENCES

- [1] C. Aggarwal, *Outlier Analysis*, Springer, 2nd edition, 2017.
- [2] 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 *DCASE. 2022*, pp. 26–30, Tampere University.
- [3] K. Dohi et al., “Description and discussion on DCASE 2023 challenge task 2: First-shot unsupervised anomalous sound detection for machine condition monitoring,” in *DCASE. 2023*, pp. 31–35, Tampere University.

- [4] J. Wang, C. Lan, C. Liu, Y. Ouyang, and T. Qin, “Generalizing to unseen domains: A survey on domain generalization,” in *IJCAI*. 2021, pp. 4627–4635, ijcai.org.
- [5] K. Wilkinghoff and F. Kurth, “Why do angular margin losses work well for semi-supervised anomalous sound detection?,” *IEEE/ACM TASLP*, vol. 32, pp. 608–622, 2024.
- [6] R. Giri, S. V. Tenneti, F. Cheng, K. Helwani, U. Isik, and A. Krishnaswamy, “Self-supervised classification for detecting anomalous sounds,” in *DCASE*, 2020, pp. 46–50.
- [7] J. A. Lopez, H. Lu, P. Lopez-Meyer, L. Nachman, G. Stemmer, and J. Huang, “A speaker recognition approach to anomaly detection,” in *DCASE*, 2020, pp. 96–99.
- [8] T. Inoue et al., “Detection of anomalous sounds for machine condition monitoring using classification confidence,” in *DCASE*, 2020, pp. 66–70.
- [9] S. Venkatesh, G. Wichern, A. Shanmugam Subramanian, and J. Le Roux, “Improved domain generalization via disentangled multi-task learning in unsupervised anomalous sound detection,” in *DCASE*. 2022, pp. 196–200, Tampere University.
- [10] T. Nishida, K. Dohi, T. Endo, M. Yamamoto, and Y. Kawaguchi, “Anomalous sound detection based on machine activity detection,” in *EUSIPCO*. 2022, pp. 269–273, IEEE.
- [11] Y. Deng et al., “Ensemble of multiple anomalous sound detectors,” in *DCASE*. 2022, Tampere University.
- [12] H. Chen et al., “An effective anomalous sound detection method based on representation learning with simulated anomalies,” in *ICASSP*. IEEE, 2023.
- [13] D. Hendrycks, M. Mazeika, and T. G. Dietterich, “Deep anomaly detection with outlier exposure,” in *ICLR*. 2019, OpenReview.net.
- [14] 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 *DCASE*, 2020, pp. 170–174.
- [15] F. Wang, J. Cheng, W. Liu, and H. Liu, “Additive margin softmax for face verification,” *IEEE SPL*, vol. 25, no. 7, pp. 926–930, 2018.
- [16] L. Ruff et al., “Deep one-class classification,” in *ICML*. 2018, vol. 80 of *Proceedings of Machine Learning Research*, pp. 4390–4399, PMLR.
- [17] P. Perera and V. M. Patel, “Learning deep features for one-class classification,” *IEEE TIP*, vol. 28, no. 11, pp. 5450–5463, 2019.
- [18] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, “A discriminative feature learning approach for deep face recognition,” in *ECCV*. Springer, 2016, pp. 499–515.
- [19] J. A. Lopez, G. Stemmer, P. Lopez-Meyer, P. Singh, J. A. del Hoyo Ontiveros, and H. A. Cordourier, “Ensemble of complementary anomaly detectors under domain shifted conditions,” in *DCASE*, 2021, pp. 11–15.
- [20] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, “ArcFace: Additive angular margin loss for deep face recognition,” in *CVPR*. 2019, pp. 4690–4699, IEEE.
- [21] 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 *DCASE*, 2021, pp. 110–114.
- [22] K. Wilkinghoff, “Combining multiple distributions based on sub-cluster adacos for anomalous sound detection under domain shifted conditions,” in *DCASE*, 2021, pp. 55–59.
- [23] K. Wilkinghoff, “Design choices for learning embeddings from auxiliary tasks for domain generalization in anomalous sound detection,” in *ICASSP*. 2023, IEEE.
- [24] X. Zhang, R. Zhao, Y. Qiao, X. Wang, and H. Li, “AdaCos: Adaptively scaling cosine logits for effectively learning deep face representations,” in *CVPR*. 2019, pp. 10823–10832, IEEE.
- [25] K. Wilkinghoff, “Sub-cluster AdaCos: Learning representations for anomalous sound detection,” in *IJCNN*. 2021, IEEE.
- [26] Q. Yu, M. S. Kavitha, and T. Kurita, “Autoencoder framework based on orthogonal projection constraints improves anomalies detection,” *Neurocomputing*, vol. 450, pp. 372–388, 2021.
- [27] L. Li, Y. Zhang, and A. Huang, “Learnable subspace orthogonal projection for semi-supervised image classification,” in *ACCV*. 2022, vol. 13843 of *Lecture Notes in Computer Science*, pp. 477–490, Springer.
- [28] A. N. Gorban, I. Yu. Tyukin, D. V. Prokhorov, and K. I. Sofeikov, “Approximation with random bases: Pro et contra,” *Information Sciences*, vol. 364-365, pp. 129–145, 2016.
- [29] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in *ICML*, 2015, vol. 37, pp. 448–456.
- [30] 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 *DCASE*, 2021, pp. 1–5.
- [31] K. Dohi et al., “MIMII DG: sound dataset for malfunctioning industrial machine investigation and inspection for domain generalization task,” in *DCASE*. 2022, pp. 26–30, Tampere University.
- [32] N. Harada, D. Niizumi, D. Takeuchi, Y. Ohishi, and M. Yasuda, “ToyADMOS2+: New toyadmos data and benchmark results of the first-shot anomalous sound event detection baseline,” in *DCASE*. 2023, pp. 41–45, Tampere University.
- [33] D. Katzman McClish, “Analyzing a portion of the ROC curve,” *Medical decision making*, vol. 9, no. 3, pp. 190–195, 1989.
- [34] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *ICLR*, Yoshua Bengio and Yann LeCun, Eds., 2015.
- [35] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *AISTATS*. 2010, vol. 9 of *JMLR Proceedings*, pp. 249–256, JMLR.org.
- [36] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, “Mixup: Beyond empirical risk minimization,” in *ICLR*. 2018, OpenReview.net.
- [37] K. Wilkinghoff, “Self-supervised learning for anomalous sound detection,” in *ICASSP*. 2024, pp. 276–280, IEEE.