On Using Pre-Trained Embeddings for Detecting Anomalous Sounds with Limited Training Data

Kevin Wilkinghoff, Fabian Fritz
Fraunhofer FKIE
Fraunhoferstraße 20, 53343 Wachtberg, Germany
{kevin.wilkinghoff, fabian.fritz}@fkie.fraunhofer.de

***Abstract*—Using embeddings pre-trained on large datasets as input representations is a popular approach for classifying audio data in case only a few training samples are available. However, for anomalous sound detection pre-trained embeddings usually perform worse than directly training a model because subtle changes indicating anomalous data are not captured sufficiently well. In this paper, the potential of using pre-trained embeddings for detecting anomalous sounds with limited training data is investigated. In experiments conducted on datasets for anomalous sound detection with domain shifts and few-shot open-set classification, it is shown that with increasing openness directly training a model on the original data leads to better performance than using pre-trained embbedings as input. Regardless of the input representation, the presented system achieves a new state-of-the-art performance for few-shot open-set classification in all pre-defined openness settings and is made publicly available.**

***Index Terms*—anomalous sound detection, domain generalization, open-set classification, few-shot learning, transfer learning, representation learning, machine listening**

I. Introduction

Semi-supervised anomalous sound detection (ASD) is the task of identifying anomalous sounds while only having access to normal data when training a system. There are several applications for ASD such as machine condition monitoring for predictive maintenance. Many recent developments have been promoted by the annual DCASE challenge [1]–[3]. For acoustic open-set classification (OSC) problems [4]–[6], normal and anomalous samples have to be distinguished but normal samples also have to be correctly classified as belonging to one of several known classes. Both tasks, ASD and OSC, are especially challenging when only few training samples are available. Examples are few-shot learning [7] for OSC with only k training samples per class (k-shot classification) [5] and ASD under domain shifts [8] between an acoustic source domain with many training samples and a target domain with only very few training samples [2], [3].

One possibility to overcome the difficulties imposed by limited training data is to use embeddings extracted with a neural network pre-trained on other very large datasets [9]. There are several ASD systems [10], [11] based on pre-trained audio embeddings and studies comparing these embeddings for ASD [12] or audio classification tasks [13] in settings with sufficient training data. However, all these systems do not achieve the same state-of-the-art performance as systems directly trained on the data. For acoustic open-set classification, the systems presented in [5], [14] use pre-trained audio embeddings. Another approach is to use image embeddings for ASD [15] or apply them for zero-shot audio classification [16]. In [17], it is shown that combining multiple hidden representations of pre-trained neural networks improves the performance. Hence, there is a substantial interest in using pre-trained embeddings for classifying audio data. Yet, evaluations in ASD settings with limited training data, which intuitively favor using pre-trained embeddings, are still missing. The goal of this work is to fill this knowledge gap.

The contributions of this work are the following. First and foremost, the ASD performance of using pre-trained audio embeddings, namely VGGish [18], openL3 [19], PANN [20] and Kumar [21] embeddings, are evaluated on the DCASE 2022 ASD dataset with domain shifts [3] and a few-shot OSC dataset [5]. It is shown that only in closed-set classification or with very few unknown classes these embeddings perform better than a model directly trained on the data whereas for more unknown classes the contrary is true even if only very few training samples are available. In conclusion, directly training a model on the data is a better approach for detecting anomalous or unknown samples. Last but not least, the proposed system[1] achieves a new state-of-the-art performance on the few-shot OSC dataset for any of the investigated input representations.

II. Pre-trained audio embeddings

For the experimental evaluations in this paper, four different audio embeddings pre-trained on large datasets have been used. These embeddings will now be briefly reviewed.

VGGish [18] is a modified version of the VGG network [22] with a similar architecture. The network is pre-trained in a supervised manner on a prelimary version of YouTube-8M [23], which consists of 2.6 billion audio segments from Youtube videos belonging to a total of 3628 classes. The resulting embeddings have a feature dimension of 128 with an additional time dimension resulting from a sliding window of 960 ms with no overlap applied to the waveforms.

OpenL3 [19] is a network trained to extract *Look, Listen, and Learn (L3)* embbedings [24], [25]. There are multiple versions of the network: One is pre-trained on a music and the other on an environmental subset of AudioSet [26] consisting

[1] An open-source implementation of the system is available at: https://github.com/wilkinghoff/few-shot-open-set-eusipco2023

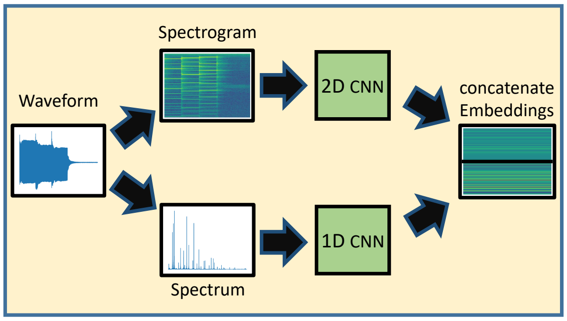

Fig. 1. Structure of the audio embedding model for direct training [28].

of 296K and 195K Youtube videos, respectively. The network is trained in a self-supervised manner to check whether a video frame and an audio clip with a length of one second do or do not belong together using an audio and a video subnetwork. After training, only the audio subnetwork is needed to extract embeddings from audio data. The resulting embeddings have a feature dimension of 512 with an additional time dimension resulting from a sliding window of one second with a hop size of 0.1 seconds applied to the waveforms.

PANN [20] is a combination of a one-dimensional subnetwork applied to waveforms (Wavegram-CNN) and a two-dimensional subnetwork applied to log-mel spectrograms. Both output representations are concatenated and further processed with another two-dimensional subnetwork. The entire network is pre-trained in a supervised manner on AudioSet [26] using a total of $1,934,187$ audio clips from Youtube videos belonging to 527 sound classes. As the difference in performance between including and not including Wavegram-CNN is relatively small, we only used the subnetwork with a VGG-like architecture pre-trained on log-mel spectrograms (CNN14). The resulting embeddings have a feature dimension of 2048 with no time dimension because of a global temporal pooling operation inside the network.

Kumar embeddings [21] are extracted using a CNN with a VGG-style architecture [22]. The network is pre-trained in a supervised manner on the balanced subset of AudioSet [26] that consists of around $22,000$ audio clips from Youtube videos belonging to 527 sound classes. The resulting embeddings have a feature dimension of 1024 and no time dimension because of a global temporal pooling operation inside the network.

III. System Descriptions

Different input representations and datasets with different tasks also require different models for further processing. These models will be described in the following subsections. All models are implemented using Tensorflow [27].

A. Anomalous sound detection systems

When directly training a model on the data, i.e. not using pre-trained embeddings, the state-of-the-art ASD system described in [28] is used. This system utilizes two different submodels that use magnitude spectrograms and magnitude spectra as input representations and are jointly trained to learn embeddings by discriminating among known classes using the sub-cluster AdaCos (scAdaCos) loss [29] with 16 sub-clusters as depicted in Fig. 1. For data augmentation, mixup [30] with a mixing coefficient drawn from a uniform distribution is used. The model is trained for 10 epochs using a batch size of 64 via adam [31]. After training the model, discriminative embeddings are extracted and compared to embeddings belonging to normal training data using cosine similarity (CS). For the source domain, k-means with 16 mean vectors is applied to obtain these normal embeddings and for the target domain, which consists of much fewer samples than the source domain, the embeddings belonging to the training samples themselves are used. Throughout the network, no bias terms and no trainable cluster centers are used as this has been shown to improve the ASD performance. Additional details about the system can be found in [28].

When using pre-trained audio embeddings as input representations, the classification model of the system is replaced with a shallow neural network for transfer learning, whose architecture is very similar to the ones used in [5], [13], [19]. More concretely, the network architecture consists of three hidden layers with dimensions of 512, 128 and 128. Prior to all other operations, all pre-trained embeddings are standardized by using batch normalization as this significantly improves the performance [19]. For the first two hidden layers ReLU is used as an activation function and batch normalization [32] is applied. The last layer does not have an activation function because it serves as the embedding layer. Prior to the last layer dropout [33] with a probability of 50% is applied. Furthermore, mixup [30] with a mixing coefficient drawn from a uniform distribution is applied to the input representations. The network is trained for 100 epochs using a batchsize of 64 by minimizing the scAdaCos loss [29] with 16 sub-clusters using adam [31]. Again, no bias terms or trainable cluster centers are used throughout the network. For the openL3 embeddings, the network pre-trained on the environmental subset has been used.

B. Few-shot open-set classification systems

For OSC, the same systems as used for ASD with only minor modifications are used. This is reasonable because the auxiliary task for training the ASD system is to discriminate among the known classes, which, in addition to detecting anomalies, is exactly the problem to be solved in OSC. All following modifications are used for both models, the model directly trained on the data and the model using pre-trained embeddings as input representations. One modification is using a single sub-cluster for each class because in a few-shot setting only very few training samples are available for each class. Another modification is using a decision threshold for identifying anomalous samples because a threshold-dependent evaluation metric is used for the experiments. A decision is derived as follows. First, the CSs between the embeddings of a test sample and the embeddings of all training samples are computed. Then, the test sample is considered anomalous if the most similar training sample is one of the known unknown samples or if the highest CS is below a fixed decision threshold. This decision threshold is set to 0.6, 0.65 and 0.75 for an openness of 0 (low), 0.04 (medium) and 0.09 (high), respectively. Following [34], openness is defined as

$$1 - \sqrt{\frac{2 \cdot C_{\text{train}}}{C_{\text{train}} + C_{\text{test}}}} \in [0, 1] \tag{1}$$

where $C_{\text{train}}, C_{\text{test}} \in \mathbb{N}$ denote the total number of classes used for training and testing, respectively. Higher openness values indicate less known or more unknown classes and a value of 0 corresponds to a closed-set classification problem. Furthermore, each model is trained using 100 times the number of shots available for training as the number of epochs and a batch size equal to eight times the number of shots. Using these adaptive hyperparameter settings depending on the size of the training dataset results in a more stable decision threshold across all openness and few-shot settings and thus improves performance. For the openL3 embeddings, the network pre-trained on the music subset has been used.

IV. Experimental results

A. Datasets

For the experimental evaluations in this paper, two different datasets have been used. The first dataset is the DCASE 2022 ASD dataset [3] for domain generalization in machine condition monitoring. The dataset consists of recordings of machines with real factory background noise each having a length of ten seconds and a sampling rate of 16 kHz, and is split into a training set, a validation set and an evaluation set. The training dataset consists of only normal recordings from seven different machine types: “fan”, “gearbox”, “bearing”, “slide rail”, “valve”, “ToyCar”, and “ToyTrain”. For each machine type, there are six different sections with 990 normal training data samples from the source domain and 10 samples from the target domain with a specific, unknown domain shift meaning that some acoustical characteristics differ for both domains. In addition to these information, there are some attribute information for each training sample that describe the state of machines or noise. Three of the sections belong to the validation set and the other three belong to the evaluation set, each having 100 normal and 100 anomalous samples belonging to the source domain and 100 normal and 100 anomalous samples belonging to the target domain. For none of these test samples, additional information such as attribute information or the domain they belong to are provided. The performance metrics for this dataset are the area under the receiver operating characteristic curve (AUC) and the partial AUC (pAUC) [35], which is the AUC for a low false positive rate ranging from 0 to 0.1 in this case. Both of these metrics are evaluated for each combination of machine type and section regardless of the domain, and the harmonic mean of all derived performance metrics is used as the final result.

TABLE I
Harmonic means of AUCs obtained with different ways to contract the temporal dimension of multiple embeddings. For PANN and Kumar embeddings, a sliding window of 960 ms has been applied to obtain a time dimension.

dataset	embeddings	mean of embeddings			mean of scores	native
		before training	during training	after training		
dev set	VGGish	**65.78 ± 0.37**	64.98 ± 0.25	58.47 ± 0.58	59.27 ± 0.58	not available
dev set	OpenL3	**70.94 ± 1.36**	70.83 ± 0.93	59.85 ± 0.49	62.67 ± 1.36	not available
dev set	PANN	64.80 ± 0.25	**66.30 ± 0.55**	59.66 ± 0.32	60.47 ± 0.17	64.21 ± 0.17
dev set	Kumar	**66.04 ± 0.76**	65.85 ± 0.83	58.94 ± 1.00	62.22 ± 0.98	60.97 ± 0.52
eval set	VGGish	**64.69 ± 0.34**	63.91 ± 0.73	58.30 ± 0.98	59.78 ± 0.53	not available
eval set	OpenL3	**69.06 ± 0.42**	68.70 ± 0.94	62.44 ± 0.46	65.02 ± 1.04	not available
eval set	PANN	63.55 ± 0.27	**65.29 ± 0.39**	58.57 ± 1.07	60.34 ± 0.62	63.33 ± 0.36
eval set	Kumar	63.56 ± 0.59	**64.05 ± 0.27**	56.95 ± 0.86	61.04 ± 0.44	60.13 ± 0.24

The second dataset is a few-shot OSC dataset [5] for acoustic alarm detection in domestic environments. The dataset consists of 24 different alarm sounds and 10 unknown sounds, namely “car horn”, “clapping”, “cough”, “door slam”, “engine”, “keyboard tapping”, “music”, “pots and pans”, “steps” and “water falling”. For each of these 34 sound classes, there are 40 different samples with a duration of four seconds and a sampling rate of 16 kHz. There are three different openness [34] settings (“low”, “medium” and “high” where training samples are provided for ten, five or none of the unknown classes, thus corresponding to an openness of 0, 0.04 and 0.09, respectively) and three different numbers of shots (one, two or four) to be used when training the OSC system. For evaluation, the dataset is divided into a different number of validation folds, depending on the number of shots to be used, by using cross-validation. When using one, two or four shots, 40, 20 or 10 validation folds are used, respectively. The performance metric for this dataset is the weighted accuracy with a weight of 0.5, which is the mean of the multiclass accuracy for the known classes and the accuracy for the unknown classes considering only the labels “known” and “unknown”.

Each experiment conducted in this paper is repeated five times and the arithmetic mean and standard deviation are determined as results. Highest values in each row of the tables containing the results are highlighted in bold letters.

B. Anomaly detection in domain-shifted conditions

Some embedding models utilize a sliding window for audio data of arbitrary length and thus consist of multiple embeddings, one for each window position. Therefore, multiple ways of combining pre-trained embeddings belonging to a single recording among the temporal axis are compared first. The results are shown in Tab. I. In contrast to the results obtained in [13], fusing the frame-wise embeddings before or during training leads to significantly better results than fusing the results after training when detecting anomalous data. Furthermore, the fact that using a sliding window for Kumar and PANN embeddings to artificially produce a time dimension improves the performance, shows that temporal structure of the original data needed to detect anomalies is not captured sufficiently well in the pre-trained embeddings.

Next, the following backends for using pre-trained embeddings as input representations are compared: 1) length normalization (LN) and a Gaussian mixture model (GMM), 2) principal component analysis (PCA), LN and a GMM, 3) linear discriminant analysis (LDA), LN and a GMM, 4) a deep neural network (DNN) with categorical cross-entropy (CXE), LN and a GMM, 5) a DNN with scAdaCos, LN and a GMM, and 6) a DNN with scAdaCos and cosine distance (CD). As shown in Tab II, for both dataset splits and all embedding types, using a shallow classifier as done in [5], [13], [19] significantly improves the performance. Moreover, using the scAdaCos loss function with CS performs best.

Last but not least, the results obtained with pre-trained embeddings are compared to directly training a model on the data as done in [28]. The results can be found in Tab. III. It can be seen that the directly trained model significantly outperforms the shallow classifiers using pre-trained audio embeddings. The most probable reason for this is that the pre-trained embeddings are not designed to and thus do not preserve subtle differences between normal and anomalous samples present in the original data. Another reason is that the recordings are very noisy, which is problematic for the embeddings that have not been exposed to the same noise conditions when being trained on the large datasets (see also [13]). A second observation to be made is that openL3 embeddings perform better than all other pre-trained embeddings, which all have a very similar performance. The most likely reason for this is that these are the only embeddings that are pre-trained in a self-supervised rather than a supervised manner. This is also consistent with the findings in [13].

C. Few-shot open-set classification

The experimental results obtained on the few-shot open-set classification dataset can be found in Tab. IV. The first observation to be made is that regardless of the system, the more shots are available for training and the lower the openness, the higher the mean performance and the smaller the variance gets. This is to be expected because more meaningful training data should always improve the results especially in settings with limited training data. Second, for all openness settings and number of shots all proposed systems outperform both baseline systems presented in [5] by a large margin and thus achieve a new state-of-the-art performance. However, there is a huge difference in performance between different input representations. On average, VGGish embeddings perform worst followed by PANN, OpenL3, Kumar and directly using the data. But interestingly, the best performing input representations have different strengths and weaknesses. OpenL3 embeddings perform best for low openness settings, which is in fact a closed-set classification task. Again, the reason could be that they are obtained by training in a self-supervised rather than a supervised manner. Directly using the data for training performs best in middle or high openness settings, which in contrast to the low openness setting include a semi-supervised ASD subtask. Kumar embeddings have a much higher performance than the system not using any embeddings in a high openness setting when using a single shot per class but perform worse in all other cases. One possible explanation could be the high variance for all performances in this training setting. A last observation to be made is that using pre-trained embeddings tends to be less severely effected when less shots are available for training than when directly using the data to train the system, which seems reasonable because this is the point of using pre-trained embeddings.

V. Conclusions

In this work, using pre-trained embeddings for ASD with limited training data has been investigated. In several experiments conducted on the DCASE 2022 ASD dataset and a recently published few-shot OSC dataset, it has been shown that directly training a model leads to better ASD performance than training a shallow classifier with pre-trained audio embeddings. On the OSC dataset, this effect was only evident for middle and high openness settings and the performance gap was not as great as for the ASD dataset. The most likely explanation is that the ASD dataset is very noisy for which pre-trained audio embeddings are known to perform worse whereas the OSC dataset is clean. Moreover, although there are only a few samples for each target domain, there are many training samples belonging to the source domains of the ASD dataset, which seem to provide enough information to also learn meaningful representations of the data in the target domains. The proposed system substantially improves upon the baseline systems of the OSC dataset thus achieves a new state-of-the-art performance and is made publicly available.

References

[1] Y. Koizumi, Y. Kawaguchi, K. Imoto, T. Nakamura, Y. Nikaido, R. Tanabe, H. Purohit, K. Suefusa, T. Endo, M. Yasuda, and N. Harada, "Description and discussion on DCASE2020 challenge task2: Unsupervised anomalous sound detection for machine condition monitoring," in *Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*, 2020, pp. 81–85.

[2] Y. Kawaguchi, K. Imoto, Y. Koizumi, N. Harada, D. Niizumi, K. Dohi, R. Tanabe, H. Purohit, and T. Endo, "Description and discussion on dcase 2021 challenge task 2: Unsupervised anomalous detection for machine condition monitoring under domain shifted conditions," in *Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*, 2021, pp. 186–190.

[3] K. Dohi, K. Imoto, N. Harada, D. Niizumi, Y. Koizumi, T. Nishida, H. Purohit, R. Tanabe, T. Endo, M. Yamamoto, and Y. Kawaguchi, "Description and discussion on DCASE 2022 challenge task 2: Unsupervised anomalous sound detection for machine condition monitoring applying domain generalization techniques," in *7th Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE).* Tampere University, 2022.

[4] A. Mesaros, T. Heittola, and T. Virtanen, "Acoustic scene classification in DCASE 2019 challenge: Closed and open set classification and data mismatch setups," in *Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE)*, 2019, pp. 164–168.

[5] J. Naranjo-Alcazar, S. Perez-Castanos, P. Zuccarello, A. M. Torres, J. J. Lopez, F. J. Ferri, and M. Cobos, "An open-set recognition and few-shot learning dataset for audio event classification in domestic environments," *Pattern Recognition Letters*, 2022.

[6] S. Shon, N. Dehak, D. A. Reynolds, and J. R. Glass, "MCE 2018: The 1st multi-target speaker detection and identification challenge evaluation," in *20th Annual Conference of the International Speech Communication Association (Interspeech).* ISCA, 2019, pp. 356–360.

[7] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Generalizing from a few examples: A survey on few-shot learning," *ACM Comput. Surv.*, vol. 53, no. 3, pp. 63:1–63:34, 2021.

[8] J. Wang, C. Lan, C. Liu, Y. Ouyang, and T. Qin, "Generalizing to unseen domains: A survey on domain generalization," in *Thirtieth International Joint Conference on Artificial Intelligence (IJCAI).* ijcai.org, 2021, pp. 4627–4635.

[9] T. Reiss, N. Cohen, L. Bergman, and Y. Hoshen, "PANDA: adapting pretrained features for anomaly detection and segmentation," in *Conference on Computer Vision and Pattern Recognition (CVPR).* Computer Vision Foundation / IEEE, 2021, pp. 2806–2814.

TABLE II

HARMONIC MEANS OF AUCS FOR DIFFERENT BACKENDS AND CONSIDERED EMBEDDINGS.

dataset	embedding	LN+GMM	PCA+LN+GMM	LDA+LN+GMM	DNN(CXE) +LN+GMM	DNN(scAdaCos) +LN+GMM	DNN(scAdaCos) +CD
dev set	VGGish	60.22 ± 0.25	60.25 ± 0.43	62.90 ± 0.13	65.40 ± 0.55	64.45 ± 0.60	**65.78 ± 0.37**
dev set	OpenL3	66.82 ± 0.19	66.33 ± 0.12	64.66 ± 0.24	68.99 ± 0.61	67.83 ± 1.24	**70.94 ± 1.36**
dev set	PANN	60.36 ± 0.09	61.48 ± 0.18	60.09 ± 0.45	63.82 ± 0.56	64.39 ± 0.75	**66.30 ± 0.55**
dev set	Kumar	61.47 ± 0.26	61.92 ± 0.29	61.82 ± 0.26	64.08 ± 1.54	64.22 ± 0.63	**65.85 ± 0.83**
eval set	VGGish	57.48 ± 0.36	57.47 ± 0.18	61.47 ± 0.20	63.77 ± 0.63	62.00 ± 1.26	**64.69 ± 0.34**
eval set	OpenL3	63.76 ± 0.23	62.62 ± 0.22	64.65 ± 0.33	67.69 ± 0.89	67.18 ± 0.43	**69.06 ± 0.42**
eval set	PANN	56.18 ± 0.13	60.08 ± 0.08	57.83 ± 1.01	61.63 ± 0.48	63.11 ± 0.48	**65.29 ± 0.39**
eval set	Kumar	60.00 ± 0.12	60.56 ± 0.13	61.56 ± 0.27	63.42 ± 0.48	61.27 ± 0.30	**64.05 ± 0.27**

TABLE III

HARMONIC MEANS OF AUCS FOR DIFFERENT INPUT REPRESENTATIONS.

dataset	VGGish	OpenL3	PANN	Kumar	no embedding [28]
dev set	65.78 ± 0.37	70.94 ± 1.36	66.30 ± 0.55	65.85 ± 0.83	**81.36 ± 0.66**
eval set	64.69 ± 0.34	69.06 ± 0.42	65.29 ± 0.39	64.05 ± 0.27	**73.43 ± 0.54**

TABLE IV

WEIGHTED ACCURACIES OBTAINED WITH DIFFERENT SYTEMS AND INPUT REPRESENTATIONS FOR VARIOUS OPENNESS SETTINGS AND NUMBER OF SHOTS PER CLASS. FOR ALL PRE-TRAINED EMBEDDINGS, INDIVIDUAL SETTINGS IDENTIFIED TO PERFORM BEST FOR ASD IN SEC. IV-B ARE USED.

openness	shots	baselines [5]		proposed system using different input representations				
		OpenL3	YAMNet	VGGish	OpenL3	PANN	Kumar	no embedding
low	1	56.8	80.1	90.0 ± 2.2	**98.1 ± 1.0**	95.6 ± 1.4	96.5 ± 1.3	97.4 ± 1.1
low	2	90.3	88.2	95.6 ± 1.6	**99.6 ± 0.3**	97.7 ± 0.8	98.5 ± 0.9	99.1 ± 0.7
low	4	97.2	94.9	98.4 ± 0.7	**99.9 ± 0.1**	98.9 ± 0.5	99.6 ± 0.4	99.7 ± 0.4
middle	1	74.1	78.3	88.7 ± 2.1	**97.0 ± 2.3**	94.9 ± 1.7	96.1 ± 1.6	96.8 ± 1.4
middle	2	86.7	85.6	93.4 ± 1.8	**99.2 ± 0.6**	95.7 ± 1.9	97.8 ± 1.3	98.7 ± 0.8
middle	4	91.3	91.9	96.2 ± 1.6	99.3 ± 0.5	97.8 ± 1.1	98.6 ± 1.3	**99.8 ± 0.2**
high	1	49.9	57.1	84.0 ± 2.6	88.8 ± 5.3	92.1 ± 2.6	**94.8 ± 2.4**	92.6 ± 4.5
high	2	58.3	61.1	87.8 ± 2.6	94.0 ± 3.2	92.9 ± 2.9	97.0 ± 1.7	**97.5 ± 1.7**
high	4	60.5	64.3	87.8 ± 2.5	96.1 ± 1.5	96.0 ± 2.1	98.4 ± 1.3	**99.1 ± 1.1**
arithmetic mean		73.9	77.9	91.3	96.9	95.7	97.5	**97.9**

[10] S. Grollmisch, D. Johnson, J. Abeßer, and H. Lukashevich, “IAEO3-combining OpenL3 embeddings and interpolation autoencoder for anomalous sound detection,” *Tech. Rep., DCASE2020 Challenge*, 2020.

[11] K. Wilkinghoff, “Using look, listen, and learn embeddings for detecting anomalous sounds in machine condition monitoring,” in *Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*, 2020, pp. 215–219.

[12] R. Müller, S. Illium, F. Ritz, and K. Schmid, “Analysis of feature representations for anomalous sound detection,” in *13th International Conference on Agents and Artificial Intelligence (ICAART)*. SCITEPRESS, 2021, pp. 97–106.

[13] S. Grollmisch, E. Cano, C. Kehling, and M. Taenzer, “Analyzing the potential of pre-trained embeddings for audio classification tasks,” in *28th European Signal Processing Conference (EUSIPCO)*. IEEE, 2020, pp. 790–794.

[14] K. Wilkinghoff, “On open-set classification with L3-net embeddings for machine listening applications,” in *28th European Signal Processing Conference (EUSIPCO)*. IEEE, 2020, pp. 800–804.

[15] R. Müller, F. Ritz, S. Illium, and C. Linnhoff-Popien, “Acoustic anomaly detection for machine sounds based on image transfer learning,” in *13th International Conference on Agents and Artificial Intelligence (ICAART)*. SCITEPRESS, 2021, pp. 49–56.

[16] D. Dogan, H. Xie, T. Heittola, and T. Virtanen, “Zero-shot audio classification using image embeddings,” in *30th European Signal Processing Conference (EUSIPCO)*. IEEE, 2022, pp. 1–5.

[17] D. Niizumi, D. Takeuchi, Y. Ohishi, N. Harada, and K. Kashino, “Composing general audio representation by fusing multilayer features of a pre-trained model,” in *30th European Signal Processing Conference (EUSIPCO)*. IEEE, 2022, pp. 200–204.

[18] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. J. Weiss, and K. W. Wilson, “CNN architectures for large-scale audio classification,” in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 131–135.

[19] A. Cramer, H. Wu, J. Salamon, and J. P. Bello, “Look, listen, and learn more: Design choices for deep audio embeddings,” in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3852–3856.

[20] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, “PANNs: Large-scale pretrained audio neural networks for audio pattern recognition,” *IEEE ACM Trans. Audio Speech Lang. Process.*, vol. 28, pp. 2880–2894, 2020.

[21] A. Kumar, M. Khadkevich, and C. Fügen, “Knowledge transfer from weakly labeled audio using convolutional neural network for sound events and scenes,” in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 326–330.

[22] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in *3rd International Conference on Learning Representations (ICLR)*, 2015.

[23] S. Abu-El-Haija, N. Kothari, J. Lee, P. Natsev, G. Toderici, B. Varadarajan, and S. Vijayanarasimhan, “Youtube-8m: A large-scale video classification benchmark,” *CoRR*, vol. abs/1609.08675, 2016. [Online]. Available: http://arxiv.org/abs/1609.08675

[24] R. Arandjelovic and A. Zisserman, “Look, listen and learn,” in *International Conference on Computer Vision (ICCV)*. IEEE Computer Society, 2017, pp. 609–617.

[25] ——, “Objects that sound,” in *15th European Conference on Computer Vision (ECCV)*, ser. Lecture Notes in Computer Science, vol. 11205. Springer, 2018, pp. 451–466.

[26] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, “Audio set: An ontology and human-labeled dataset for audio events,” in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 776–780.

[27] M. Abadi *et al.*, “Tensorflow: A system for large-scale machine learning,” in *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2016, pp. 265–283.

[28] 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 (ICASSP)*. IEEE, 2023.

[29] ——, “Sub-cluster AdaCos: Learning representations for anomalous sound detection,” in *International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2021.

[30] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, “Mixup: Beyond empirical risk minimization,” in *6th International Conference on Learning Representations (ICLR)*, 2018.

[31] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *3rd International Conference on Learning Representations (ICLR)*, 2015.

[32] S. Ioffe and C. Szegedy, “Batch normalization: accelerating deep network training by reducing internal covariate shift,” in *32nd International Conference on Machine Learning (ICML)*, vol. 37, 2015, pp. 448–456.

[33] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Improving neural networks by preventing co-adaptation of feature detectors,” *CoRR*, vol. abs/1207.0580, 2012.

[34] W. J. Scheirer, A. de Rezende Rocha, A. Sapkota, and T. E. Boult, “Toward open set recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1757–1772, 2013.

[35] D. K. McClish, “Analyzing a portion of the ROC curve,” *Medical decision making*, vol. 9, no. 3, pp. 190–195, 1989.