applied sciences

MDPI

*Article*

# Unsupervised Anomalous Sound Detection for Machine Condition Monitoring Using Classification-Based Methods

**Yaoguang Wang [1,†], Yaohao Zheng [2,†], Yunxiang Zhang [2,†], Yongsheng Xie [3], Sen Xu [3], Ying Hu [2] and Liang He [1,2,*]**

[1] Department of Electronic Engineering, Tsinghua University, Beijing 100084, China; wyg18@mails.tsinghua.edu.cn
[2] School of Information Science and Engineering, Xinjiang University, Urumqi 830046, China; zhangyunxiang0128@163.com (Y.Z.); 17801007427@163.com (Y.Z.); huying@xju.edu.cn (Y.H.)
[3] State Grid Xinjiang Electric Power Company, Urumqi 830092, China; 18099622921@189.cn (Y.X.); 551677089@163.com (S.X.)
* Correspondence: heliang@mail.tsinghua.edu.cn; Tel.: +86-0991-8582510
† These authors contributed equally to this work.

**Abstract:** The task of unsupervised anomalous sound detection (ASD) is challenging for detecting anomalous sounds from a large audio database without any annotated anomalous training data. Many unsupervised methods were proposed, but previous works have confirmed that the classification-based models far exceeds the unsupervised models in ASD. In this paper, we adopt two classification-based anomaly detection models: (1) Outlier classifier is to distinguish anomalous sounds or outliers from the normal; (2) ID classifier identifies anomalies using both the confidence of classification and the similarity of hidden embeddings. We conduct experiments in task 2 of DCASE 2020 challenge, and our ensemble method achieves an averaged area under the curve (AUC) of 95.82% and averaged partial AUC (pAUC) of 92.32%, which outperforms the state-of-the-art models.

**Keywords:** unsupervised anomalous sound detection; classification-based model; Outlier classifier; ID classifier



**Citation:** Wang, Y.; Zheng, Y.; Zhang, Y.; Xie, Y.; Xu, S.; Hu, Y.; He, L. Unsupervised Anomalous Sound Detection for Machine Condition Monitoring Using Classification-Based Methods. *Appl. Sci.* **2021**, *11*, 11128. https://doi.org/10.3390/app112311128

Academic Editor: Francesc Alías

Received: 21 October 2021
Accepted: 7 November 2021
Published: 24 November 2021



## 1. Introduction

Anomalous sound detection (ASD) is the task to identify whether the sound is normal or anomalous. This technique is commonly used in audio surveillance [1,2], machine condition monitoring [3], medical diagnosis [4], smart city construction [5], etc. In the case of machine condition monitoring, we hope to monitor the operation of the machine through acoustic characteristics, because sound-based anomaly detection is flexible and the cost can be reduced by bringing the microphone close to different machines to detect anomalies. It can avoid the huge loss caused by serious failure that find out the early fault of the machine and carry out maintenance effectively.

ASD based on Machine Learning algorithms includes supervised-ASD and unsupervised-ASD. For supervised-ASD, the training data contains both normal and anomalous sounds as shown in Figure 1a, the supervised binary classification model is suitable for anomaly detection. Since the machine works normally most of the time, it is difficult to collect a large number of anomalous sounds, and the pattern of anomalous sounds emitted from a target machine is not clear. Only normal sounds are provided as training data as shown in Figure 1b, which makes ASD an unsupervised task. The "Unsupervised Detection of Anomalous Sounds for Machine Condition Monitoring" task of Detection and Classification of Acoustic Scenes and Events 2020 (DCASE 2020) [6], this task is mainly to detect whether the sound emitted by the machine is normal or abnormal based on the unmarked data set provided during the operation of various machines, has attracted many researchers to submit systems, and their systems ranked on public data sets [7,8]. The data used for this task comprises parts of ToyADMOS and the MIMII Dataset consisting of the

normal/anomalous operating sounds of six types of toy/real machines. Each recording is a single-channel (approximately) 10-s length audio that includes both a target machine's operating sound and environmental noise. The expected goal of this paper is to establish a classification model based on unsupervised learning to detect abnormal sounds on the data set given by the task, and the results are better than all the models submitted before.

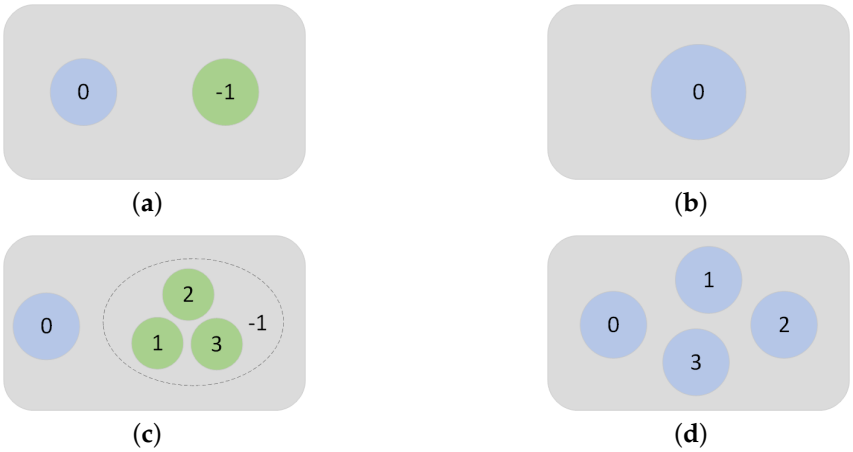



**Figure 1.** Different data settings. (**b**) stands for unsupervised settings, (**a**,**c**,**d**) represent three forms of supervised setting respectively. Blue area denotes normal samples, green denotes anomalous or outlier samples, grey denotes unavailable samples.In Figure 1a, the marked normal data and abnormal data are given, which belongs to the supervised learning method. In Figure 1b, only normal samples are given, and no abnormal samples are given. In Figure 1c, marked normal samples and unmarked abnormal samples are given. In Figure 1d, the labeled normal samples are given, and the abnormal samples are not given. The above three categories belong to unsupervised learning.

Some methods use unsupervised models to learn the essential characteristics of normal sounds so that find the subspace where the normal samples are located, and the sounds outside the subspace are judged as anomalous. Koizumi et al. [6] adopts an autoencoder as the anomaly detector, the model is trained with reconstruction error on normal samples and the anomaly scores are derived from the reconstruction error. Hayashi et al. [9] proposed an improved method for unsupervised abnormal sound detection based on autoencoder. This method uses the self-attention architecture based on Transform and Conformer. Different from the standard automatic decoder, this method can extract sequential level information from the entire audio input, significantly improving the performance of ASD. An x-vector based model using L3-Net embeddings for anomalous sound detection has been proposed in [10]. L3-net consists of two convolutional neural networks, video subnet and audio subnet. In this article, only its audio subnet is used, and the open source implementation openl3 pre trained on the music subset of audioset is used to extract L3 net embedding. However, the network forces the differentiated behavior of x-vector, resulting in performance degradation. Durkota et al. [11] combines the Siamese Network feature extractor with KNN anomaly detector, the Siamese Network extracts required features and then the KNN trained on the features performs anomaly detection. KNN (k-nearest neighbor algorithm), that is, given a training data set, for a new input instance, find the K instances closest to the instance in the training data set. This method does not need to establish a model, but it has a large amount of calculation. For each sample to be classified, it is necessary to calculate its distance to all known samples in order to obtain its K nearest neighbors, and it is easy to ignore a small number of samples. Haunschmid et al. [12] adopts Masked Autoregressive Flows (MAFs) to learn the density of normal sounds and uses the negative log-likehood as the anomaly score. MAFs is a neural density estimator, which is the best proposed for tasks where evaluating densities is more important than generating new data. Compared with similar structures, MAFs has the advantage of fast likelihood estimation. Some works have demonstrated that the use of machine ID

information significantly improves the ASD performance [13–17]. In Figure 1c,d, data sets from other machine IDs are added to the training data. Ref. [15] divides the training data into two categories, the normal sounds of a specific machine ID are regarded as positive samples, and the normal sounds of other machines IDs are considered as negative samples. In [13,16], the authors treat the different machine IDs as different categories, and in [14], the authors adds anomalous samples through data augmentation.

In this paper, we adopt two methods for anomaly detection. The first method is to train an Outlier classifier based on Figure 1c setting. The model distinguishes anomalies from the normal, and its output is directly used as the anomaly score of the unseen sound. Another method trains an ID classifier based on Figure 1d setting, its output is the probability that the unseen sound belongs to the corresponding machine ID, and its opposite number is taken as the anomaly score. At the same time, we calculate the similarity of embeddings between the normal sounds and the unseen sounds for anomaly detection.

The rest of this paper is organized as follows. In Section 2, in this section, we establish a model based on the classification method to obtain the decision boundary to judge whether the invisible sound is normal or abnormal. In Section 3, we report the results of experiments conducted to evaluate the performance of unsupervised anomalous sound detection. We then conclude this paper in Section 4.

## 2. Proposed Method

The research described in [13–17] shows that the supervised classifier substantially outperforms the unsupervised methods across most machine types in anomalous sound detection. The difference between supervised learning and unsupervised learning is whether the data set provided has been marked. Supervised learning refers to learning a function from a given labeled training data set, which is the most common classification problem, while unsupervised learning refers to learning the statistical law or internal structure of data from unlabeled data. In these works, unsupervised anomaly detection is reframed as a supervised classification problem. CNN (Concolutional Neural Network) has demonstrated its good performance for audio classification, such as ResNet [18], MobileFaceNet [19], MobileNetV3 [20]. Both of them are based on CNN network structure. ResNet is proposed to better solve the problem of gradient disappearance and explosion with the deeper and deeper layers of the deep network. Its prominent feature is that it puts forward the connection mode of shortcut connection for the first time. MobileFaceNet and mobilenetv3 belong to lightweight neural networks, of which MobileFaceNet is an improved version of MobileNetV2 .The residual block structure in ResNet is used for reference to further compress the network model, and the separable layer is used to replace the average pooling layer, which solves the problem that it is difficult to converge when using the global average pooling layer in MobileNetV2. MobileNetV3 is a model obtained by architecture search, which draws lessons from the deep separable convolution in MobileNetV1 and the inverse residual structure of linear bottleneck in MobileNetV2, The swish-x activation function is introduced, which further improves the accuracy and reduces the delay compared with MobileNetV2. The model established in this article draws lessons from the above structure and is inspired by [21], in which CBAM (convolutionary block attention module) is introduced. There are some differences between classifiers, which will lead to different classification boundaries, that is, there may be errors. Then, after merging multiple classifiers, we can get a more reasonable boundary, reduce the overall error rate and achieve better results. In this section, we adopt two classifiers based on above popular architectures to obtain decision boundary for identifying whether the unseen sound is normal or anomalous. We show their network structure in Figure 2.

In the experiment, we use the audio processing library Liborsa to process the audio data [22], and use the warm restarts gradient descent method to solve the local optimal problem that is prone to occur during gradient descent [23].

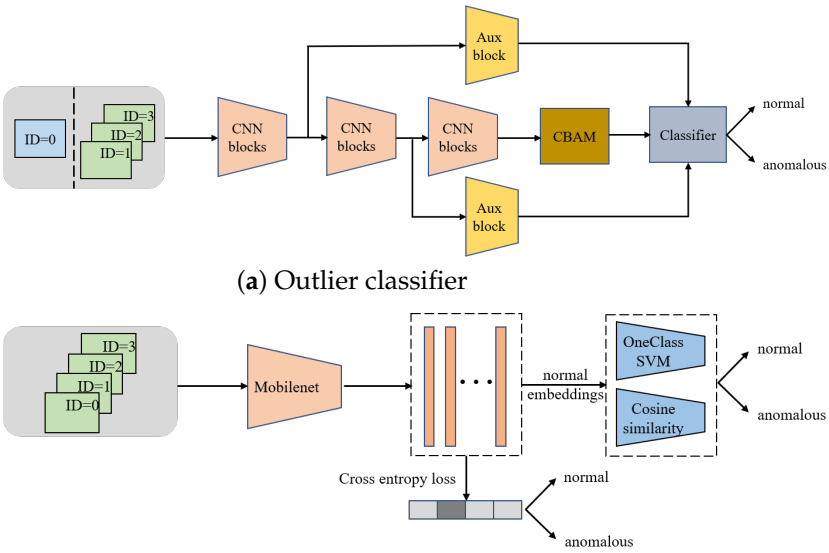



**Figure 2.** (**a**) The Outlier classifier distinguishs the outliers which are considered as anomalies from the normal, and it directly outputs the anomaly scores of unseen sounds. (**b**) The ID classifier identifies different machine IDs. For the ID classifier, we use two methods for anomaly detection as shown in Figure 2b, the first is to calculate the similarity between unseen sounds and normal sounds using embeddings extracted from the hidden layer, another method uses the confidence of classification.

### *2.1. Outlier Classifier for Binary Classification*

In order to solve anomaly detection problem in a supervised manner, we obtain training set containing normal and anomalous samples according to Figure 1c, which is an unlabeled data set. For each specific machine ID, we assign the audio clips of this machine ID as positive samples and the other machine IDs in the same domain as negative samples.

#### 2.1.1. Attention-Based Audio Classification Network

Primus et al. [15] adopts this network in anomalous sound detection by changing the filters sizes slightly and outperforms the most methods across all machine types and IDs. In this paper, we add Convolutional Block Attention Module (CBAM) [21] which contains of Channel-attention module (CAM) and Spatial-attention module (SAM), the specific structure of CBAM is shown in Figure 3, they are concerned about "*what*" and "*where*" the audio events happen respectively. CAM can be regarded as a process of selecting relevant semantic features based on context semantics. When the network wants to predict the "*fan*" audio, CAM will assign larger weight to the feature map containing the "*fan*" spectrum structure. The SAM will locate the segments of "*fan*" on the feature map, thereby filtering out background noise. So attention module is helpful for accurately expressing the characteristics of normal sounds.

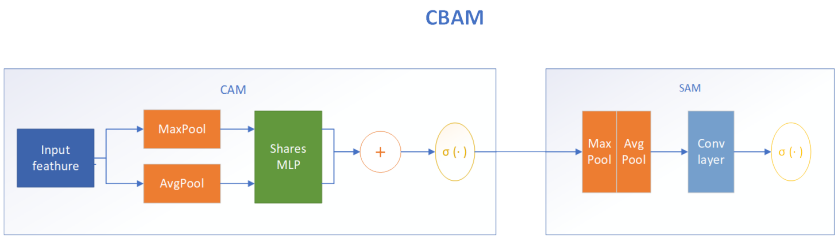



**Figure 3.** The Convolutional Block Attention Module (CBAM) contains of Channel-attention module (CAM) and Spatial-attention module (SAM).

The feature map $\mathbf{X}$ ($C \times H \times W$) passes through CAM and then SAM. CAM calculates the weight ($C \times 1 \times 1$) of each channel, and multiplies the weight with the original feature map to obtain a weighted feature map. In order to obtain the weight of the channel dimension, this module calculates the average value and maximum value of each channel respectively with *avgpool* and *maxpool*, and feeds them to a common multi-layer perceptron, and then the two outputs are added together and normalized by the sigmoid function to get the final weight. CAM is defined as:

$$\begin{aligned} \mathbf{W}_C = \sigma(\mathbf{W}_2(\delta(\mathbf{W}_1 \cdot avgpool(\mathbf{X}))) \\ + \mathbf{W}_2(\delta(\mathbf{W}_1 \cdot maxpool(\mathbf{X})))) \end{aligned} \tag{1}$$

where $\mathbf{W}_1 \in \mathbb{R}^{C \times \frac{C}{r}}$, $\mathbf{W}_2 \in \mathbb{R}^{\frac{C}{r} \times C}$ represent FC layers, $\delta(\cdot)$, $\sigma(\cdot)$ represent ReLU and sigmoid function respectively, $r$ denotes the scaling ratio.

SAM calculates the average and maximum values of different channels on the same point to obtain weights ($1 \times H \times W$) with $avgpool'$ and $maxpool'$, concatenates them along the channel dimensions, and then the weights passes a convolutional layer and sigmoid function to get the final weights. The weights is multiplied by each channel on the time-frequency domain to obtain a weighted feature map. SAM is defined as:

$$\mathbf{W}_S = \sigma\big(\mathbf{W}\big[avgpool'(\mathbf{X}); maxpool'(\mathbf{X})\big]\big) \tag{2}$$

where $\mathbf{W}$ denotes a convolutional layer. CAM and SAM are connected in a sequential manner, the order of operations is as follows:

$$\begin{aligned} \mathbf{Y} &= \mathbf{W}_C(\mathbf{X}) \otimes \mathbf{X} \\ \mathbf{Z} &= \mathbf{W}_S(\mathbf{Y}) \otimes \mathbf{Y} \end{aligned} \tag{3}$$

where $\otimes$ represents element-wise multiplication.

#### 2.1.2. Auxiliary Classifiers for Anomaly Detection

Since we have defined the outlier data of normal sounds as the anomalous, the outputs of the classifier are used as the anomaly scores. The network composes of multiple convolution blocks as shown in Figure 2a. The front stages have a larger kernel size and more pooling operations to reduce the feature dimension, while the back stages have a smaller kernel size and fewer pooling operations to maintain the resolution of the features, thereby limiting the receptive fields to capture local features [24]. In order to improve the classification ability of the network, we adopt the strategy of auxiliary classification. Each stage is followed by an auxiliary classifier, and a CBAM module is added in the last stage. We use multiple-level features at the same time by integrating the outputs of auxiliary classifiers according to the weights, where the back classifiers have greater weights,

$$p = (w_1 \cdot p_1 + w_2 \cdot p_2 + w_3 \cdot p_3) \tag{4}$$

where $w_i$, $p_i$ ($i = 1, 2, 3$) denote the weight and the output of the $i$-th classifier respectively, $p$ denotes the final output of the network and is used as the anomaly score. We believe that the deeper the features, the stronger the expressiveness and the higher the accuracy of classification. In Equation (4), $w_1 < w_2 < w_3$. The specific weight value is set based on the training set according to the trust degree.

### *2.2. ID Classifier for Multiple Classification*

We train an ID Classifier to recognize different machine IDs of the same machine type with recordings from all the machine IDs. The model uses the embeddings output by the hidden layer of the model to determine whether the audio is anomalous, and uses the classification confidence of the network to identify anomalies.

#### 2.2.1. MobileNet-Based Audio Classification Network

In this section, we introduce a model that combines the characteristics of MobileFaceNet [19] and MobileNetV3 [20]. We adopt MobileNetV3 as the main body of the network structure and modify the network parameters as shown in Table 1.

**Table 1.** Architecture of MobileNet.

| Operator | Exp Size | *#out* | SE | NL | *s* |
|---|---|---|---|---|---|
| conv3 × 3 | - | 32 | - | HS | 2 |
| bneck3 × 3 | 64 | 32 | - | RE | 1 |
| bneck3 × 3 | 64 | 32 | - | RE | 2 |
| bneck3 × 3 | 64 | 32 | - | RE | 1 |
| bneck3 × 3 | 64 | 32 | ✓ | RE | 2 |
| bneck3 × 3 | 64 | 32 | ✓ | RE | 1 |
| bneck3 × 3 | 128 | 64 | ✓ | RE | 1 |
| bneck3 × 3 | 128 | 64 | - | HS | 2 |
| bneck3 × 3 | 128 | 64 | - | HS | 1 |
| bneck3 × 3 | 128 | 64 | - | HS | 1 |
| bneck3 × 3 | 128 | 64 | - | HS | 1 |
| bneck3 × 3 | 256 | 128 | ✓ | HS | 1 |
| bneck3 × 3 | 256 | 128 | ✓ | HS | 1 |
| bneck3 × 3 | 256 | 128 | ✓ | HS | 1 |
| bneck3 × 3 | 256 | 128 | ✓ | HS | 2 |
| bneck3 × 3 | 256 | 128 | ✓ | HS | 1 |
| conv1 × 1 | - | 512 | - | HS | 1 |
| GDConv32 × 1 | - | 512 | - | - | 1 |
| conv1 × 1 | - | 128 | - | - | 1 |

*#out* refers to the number of out channels, SE refers to Squeeze-And-Excite block, HS refers to h-swish, RE refers to ReLU, *s* refers to stride, and NL refers to nonlinear.

The model inherits the advantages of MobileNetV3. Depthwise separable convolutions contain spatial filtering and feature generation, which has fewer parameters and lower computational cost compared with conventional convolution. The linear bottleneck and inverted residual structure map features into high-dimensional space to increase the expressiveness of the network. The squeeze and excitation is integrated as attention module. We use h-swish or ReLU as the non-linearity. We also use global depthwise convolution (GDConv) to replace global pooling like MobileFaceNet.

#### 2.2.2. Anomaly Detection in Multiple Ways

For the ID classifier, we use two methods for anomaly detection as shown in Figure 2b. The first method is to use the embeddings output by the hidden layer of the network to calculate the similarity between the unseen sound and the normal sound, and the similarity is calculated in two ways, we will present the detailed introduction of the two calculation methods in the results section. Another method uses the softmax probability output by the network as the probability that the sample belongs to the corresponding machine ID, and its opposite number is used as the anomaly score. We apply different methods on different machines.

## 3. Results

The two trained models have different definitions for anomaly detection. The Outlier classifier is trained for distinguishing anomalies from normal sounds, so the outputs of the model are directly used as the anomaly scores. We also apply the same supervised settings shown in Figure 1c as the Outlier classifier to the network in Figure 2b, but it doesn't perform well. Different from the Outlier classifier, we train the ID Classifier to recognize different IDs of the same machine type and learn the hidden characteristics of the

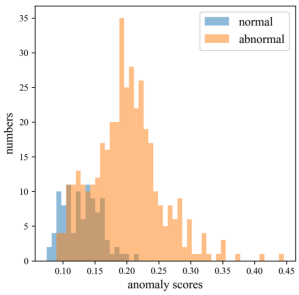
normal sounds. We calculate the similarity between the embeddings of unseen sounds and corresponding normal sounds for anomaly detection in two ways: angle (Cosine similarity) and distance (OneClassSVM), and the final anomaly score is calculated as "1-similarit". It is worth noting that OneClassSVM is suitable for anomaly detection of the machine "*ToyCar*", Cosine similarity is suitable for other machine types according to our experiments.

The comparison of our methods against other advanced approaches on the evaluation set of DCASE 2020 task 2. The evaluation indexes we use here are AUC (area under curve) and pAUC (partial area under curve), where AUC refers to the area under the ROC curve, and pAUC refers to the area under the ROC curve within the false positive rate [0, P]. Comparison results with existing models in DCASE are shown in Table 2, we can find our methods performs well on different machines, the Outlier classifier achieves the average AUC of 93.97% and average pAUC of 89.75% and the ID classifier achieves the average AUC of 92.09% and average pAUC of 87.81%. We summarize the advanced systems on DCASE 2020 task 2 into two categories: the classification-based models [13–17] and unsupervised-based models [9–12], of these, the modle of Giri [13] based on multi-class classification ranked first in task 2 of DCASE2020. The unsupervised-based models like autoencoder, principal component analysis (PCA), KNN and normalizing flow only use normal sounds of the target machine as the training set. The classification-based models add another data sets to create a training set includeing multiple categories, and convert unsupervised anomaly detection to supervised or semi-supervised tasks. We can see that the classification-based models outperform the unsupervised-based models by a large margin, outlier samples can greatly help the model to recognize anomalous sounds. The experimental results confirm that the machine ID information is beneficial to accurately determine the classification boundary of the classifier and extract more distinguishing hidden features.

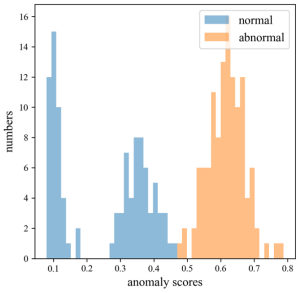
Table 2 shows that even the best models cannot perform best on all machine types and the performance of different machines of the same type varies greatly as shown in Figure 4. So we apply the model ensemble strategy. For the target machine, we choose the model with better performance on development data set. Our ensemble method achieves the highest average AUC of 95.82% and average pAUC of 92.32%, even outperforms all other methods on some machine types such as "*fan*".

**Table 2.** AUC (%) and pAUC (%) for each machine

| | **Fan** | **Pump** | **Slider** | **Valve** | **Toy-Car** | **Toy-Conveyor** | **Average** |
|---|---|---|---|---|---|---|---|
| | **AUC(pAUC)** | **AUC(pAUC)** | **AUC(pAUC)** | **AUC(pAUC)** | **AUC(pAUC)** | **AUC(pAUC)** | **AUC(pAUC)** |
| Baseline [6] | 82.80(65.80) | 82.37(64.11) | 79.41(58.87) | 57.37(50.79) | 80.14(66.17) | 85.36(66.95) | 77.91(62.12) |
| Hayashi [9] | 92.72(80.52) | 90.63(73.61) | 95.68(81.48) | 97.43(89.69) | 91.75(83.97) | **92.10** (76.76) | 93.39(81.01) |
| Wilkinghoff [10] | 93.75(80.68) | 93.19(81.10) | 95.71(79.45) | 94.87(83.58) | 94.06(86.80) | 84.22(69.12) | 92.63(80.12) |
| Durkota [11] | 90.74(83.38) | 88.70(75.97) | 96.18(87.49) | 97.48(92.46) | 94.32(89.01) | 64.38(53.79) | 88.63(80.35) |
| Haunschmid [12] | 91.48(74.32) | 92.30(72.14) | 89.74(76.43) | 81.99(69.82) | 81.50(67.00) | 88.01(70.52) | 87.50(71.71) |
| Giri [13] | 94.54(84.30) | 93.65(81.73) | 97.63(89.73) | 96.13(90.89) | 94.34(89.73) | 91.19(73.34) | 94.58(84.95) |
| Daniluk [14] | 99.13(96.40) | 95.07(90.23) | 98.18(91.98) | 90.97(77.41) | 93.52(83.87) | 90.51(77.56) | 94.56(86.24) |
| Primus [15] | 96.84(95.24) | **97.76**(92.24) | 97.29(88.74) | 90.15(86.65) | 86.37(83.83) | 88.28(79.15) | 92.78(87.64) |
| Inoue [16] | 98.84(94.89) | 94.37(88.27) | 95.68(83.09) | **97.82**(**94.93**) | 93.16(87.69) | 87.41(72.03) | 94.55(86.82) |
| Zhou [17] | 99.79(98.92) | 95.79(**92.60**) | 99.84(99.17) | 91.83(84.74) | **95.60**(**91.30**) | 73.61(64.06) | 92.74(88.47) |
| Outlier classifier | 97.53(95.64) | 97.34(91.54) | 99.04(95.14) | 92.00(89.05) | 88.11(86.53) | 89.80(**80.61**) | 93.97(89.75) |
| ID classifier | 99.94(99.80) | 95.01(90.89) | 99.09(95.91) | 95.82(93.58) | 91.33(86.57) | 71.32(60.09) | 92.09(87.81) |
| ensemble | **99.96**(**99.84**) | 97.35(91.58) | **99.97**(**99.83**) | 95.82(93.58) | 92.02(88.50) | 89.80(**80.61**) | **95.82**(**92.32**) |




(**a**) "*Valve*" ID : 5




(**b**) "*Valve*" ID : 1

**Figure 4.** Distribution of anomaly scores of the machine "*Valve*". (**a**,**b**) are different machines of the same types. (**b**) shows that the model completely distinguishes normal and anomalous sounds, but (**a**) does not.

## 4. Conclusions

In this paper, we introduce two classification-based models for the anomaly detection and conduct experiments in task 2 of DCASE 2020 challenge. Both models are trained with only normal sounds to learn the distribution characteristics of the normal sounds like most unsupervised methods, and then the unseen sounds are identified as the anomalous when they are outliers of normal sounds. The practical application of the model is mainly aimed at automatic machine fault detection, which is a key technology of the fourth industrial Revolution including factory automation based on artificial intelligence. The abnormal machine can be recognized by the sound of the machine running to speed up the process of industrial automation. In normal factories, it is extremely difficult to obtain the abnormal sound of machine operation. The model mentioned in this paper can identify abnormal samples in the data with a large number of normal samples and realize automatic detection of machine faults. We will continue to conduct in-depth research on this. Different from the unsupervised methods, we also use samples from other machine IDs to train the models in a supervised manner, so that the classification-based method can be used to find the decision boundary between the normal and outliers. The use of machine ID information helps to determine the decision boundary accurately and improve the ASD performance. At present, our work only achieves the best average accuracy of anomaly detection for six machine types, but does not achieve the best effect on all machine types. Further improving the generalization ability of the model will be our next improvement direction Table 2 demonstrates that the classification-based models outperform the unsupervised-based models significantly across all machine types, and our models outperform the state-of-the-art models, achieving an averaged AUC of 95.82% and an averaged pAUC of 92.32% with an ensemble strategy.

**Author Contributions:** Conceptualization, Y.W. and L.H.; methodology, Y.W., Y.Z. (Yunxiang Zhang) and Y.Z. (Yaohao Zheng); software, Y.W., Y.Z. (Yunxiang Zhang) and Y.Z. (Yaohao Zheng); validation, L.H., Y.W., Y.Z. (Yaohao Zheng) and Y.Z. (Yunxiang Zhang).; formal analysis, Y.W., Y.Z. (Yunxiang Zhang) and Y.Z. (Yaohao Zheng); investigation, Y.W., Y.Z. (Yunxiang Zhang) and Y.Z. (Yaohao Zheng); resources, L.H., Y.X., Y.H. and S.X.; data curation, Y.W. and L.H.; writing—original draft preparation, Y.W. and L.H.; writing—review and editing, Y.Z. (Yunxiang Zhang), Y.Z. (Yaohao Zheng) and L.H.; visualization, Y.W., Y.Z. (Yaohao Zheng) and Y.Z. (Yunxiang Zhang); supervision, L.H.; project administration, L.H. and Y.W.; funding acquisition, L.H. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable

**Data Availability Statement:** Publicly available datasets were analyzed in this study. This data can be found here: [http://dcase.community/challenge2020/ (accessed on 1 October 2021)].

**Conflicts of Interest:** The authors declare no conflict of interest.

## References


1. Foggia, P.; Petkov, N.; Saggese, A.; Strisciuglio, N.; Vento, M. Audio Surveillance of Roads: A System for Detecting Anomalous Sounds. *IEEE Trans. ITS* **2016**, *17*, 279–288. [CrossRef]
2. Ntalampiras, S.; Potamitis, I.; Fakotakis, N. Probabilistic Novelty Detection for Acoustic Surveillance Under Real-World Conditions. *IEEE Trans. Multimed.* **2011**, *13*, 713–719. [CrossRef]
3. Koizumi, Y.; Saito, S.; Uematsu, H.; Harada, N. Optimizing Acoustic Feature Extractor for Anomalous Sound Detection Based on NeymanPearson Lemma. In Proceedings of the 2017 25th European Signal Processing Conference (EUSIPCO), Kos, Greece, 28 August–2 September 2017.
4. Zheng, X.; Zhang, C.; Chen, P.; Zhao, K.; Jia, W. A CRNN System for Sound Event Detection Based on Gastrointestinal Sound Dataset Collected by Wearable Auscultation Devices. *IEEE Access* **2020**, *8*, 157892–157905. [CrossRef]
5. Ick, C.; Mcfee, B. Sound Event Detection in Urban Audio With Single and Multi-Rate PCEN. *arXiv* **2021**, arXiv:2102.03468.
6. Koizumi, Y.; Kawaguchi, Y.; Imoto, K.; Nakamura, T.; Nikaido, Y.; Tanabe, R.; Purohit, H.; Suefusa, K.; Endo, T.; Yasuda, M.; et al. Description and discussion on DCASE2020 challenge task2: Unsupervised anomalous sound detection for machine condition monitoring. *arXiv* **2020**, arXiv:2006.05822.
7. Koizumi, Y.; Saito, S.; Uematsu, H.; Harada, N.; Imoto, K. ToyADMOS: A dataset of miniature-machine operating sounds for anomalous sound detection. In Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), New Paltz, NY, USA, 20–23 October 2019; pp. 308–312.
8. Purohit, H.; Tanabe, R.; Ichige, T.; Endo, T.; Nikaido, Y.; Suefusa, K.; Kawaguchi, Y. MIMII Dataset: Sound dataset for malfunctioning industrial machine investigation and inspection. In Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019), Tokyo, Japan, 20 September 2019; pp. 209–213.
9. Hayashi, T.; Toshimura, T.; Adachi, Y. *Conformer-Based ID-Aware Autoencoder for Unsupervised Anomalous Sound Detection*; Tech. Report in DCASE2020 Challenge Task 2; Detection and Classification of Acoustic Scenes and Events: Nagoya, Japan, 2020.
10. Wilkinghoff, H. *Anomalous Sound Detection with Look, Listen, and Learn Embeddings*; Tech. Report in DCASE2020 Challenge Task 2; Detection and Classification of Acoustic Scenes and Events: Munich, Germany, 2020.
11. Durkota, K.; Linda, M.; Ludvik, M.; Tozicka, J. *Euron-Net: Siamese Network for Anomaly Detection*; Tech. Report in DCASE2020 Challenge Task 2; Detection and Classification of Acoustic Scenes and Events: Prague, Czech Republic, 2020.
12. Haunschmid, V.; Praher, P. *Anomalous Sound Detection with Masked Autoregressive Flows and Machine Type Dependent Postprocessing*; Tech. Report in DCASE2020 Challenge Task 2; Detection and Classification of Acoustic Scenes and Events 2020: Linz, Austria, 2020.
13. Giri, R.; Tenneti, S.V.; Cheng, F.; Helwani, K.; Isik, U.; Krishnaswamy, A. *Unsupervised Anomalous sound Detection Using Self-Supervised Classification and Group Masked Autoencoder for Density Estimation*; Tech. Report in DCASE2020 Challenge Task 2; Detection and Classification of Acoustic Scenes and Events: Palo Alto, CA, USA, 2020.
14. Daniluk, P.; Goździewski, M.; Kapka, S.; Kośmider, M. *Ensemble of Auto-Encoder Based and WaveNet Like Systems for Unsupervised Anomaly Detection*; Tech. Report in DCASE2020 Challenge Task 2; Detection and Classification of Acoustic Scenes and Events: Warsaw, Poland, 2020.
15. Primus, P. *Reframing Unsupervised Machine Condition Monitoring as a Supervised Classification Task with Outlier Exposed Classifiers*; Tech. Report in DCASE2020 Challenge Task 2; Detection and Classification of Acoustic Scenes and Events: Linz, Austria, 2020.
16. Inoue, T.; Vinayavekhin, P.; Morikuni, S.; Wang, S.; Trong, T.H.; Wood, D.; Tatsubori, M.; Tachibana, R. *Detection of Anomalous Sounds for Machine Condition Monitoring Using Classification Confidence*; Tech. Report in DCASE2020 Challenge Task 2; Detection and Classification of Acoustic Scenes and Events: Tokyo, Japan, 2020.
17. Zhou, Q. *ArcFace Based Sound MobileNets for DCASE 2020 Task 2*; Tech. Report in DCASE2020 Challenge Task 2; Detection and Classification of Acoustic Scenes and Events: Shanghai, China, 2020.
18. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
19. Chen, S.; Liu, Y.; Gao, X.; Han, Z. MobileFaceNets: Efficient CNNs for accurate realTime face verification on mobile devices. In Proceedings of the 13th Chinese Conference, CCBR 2018, Urumqi, China, 11–12 August 2018.
20. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for MobileNetV3. In Proceedings of the International Conference on Computer Vision 2019 (ICCV2019), Seoul, Koren, 27 October–2 November 2019.
21. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the 15th European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
22. McFee, B.; Raffel, C.; Liang, D.; Ellis, D.P.W.; McVicar, M.; Battenberg, E.; Nieto, O. Librosa: Audio and Music Signal Analysis in Python. In Proceedings of the 14th Python in Science Conference, New York, NY, USA, 6–12 August 2015; pp. 18–25.
23. Loshchilov, I.; Hutter, F. SGDR: Stochastic gradient descent with warm restarts. *arXiv* **2016**, arXiv:1608.03983.
24. Koutini, K.; Eghbal-zadeh, H.; Widmer, G. Receptivefield-regularized CNN variants for acoustic scene classification. *arXiv* **2019**, arXiv:1909.02859.