CNN-BASED ANOMALOUS SOUND DETECTION SYSTEM FOR DOMAIN GENERALIZATION

Technical Report

Hiroki Narita

Aichi Institute of Technology
Graduate School of Business
Administration and Computer Science
Aichi, Japan
hirokin1999@outlook.jp

Akira Tamamori

Aichi Institute of Technology
Department of Information Science
Aichi, Japan
akira-tamamori@aitech.ac.jp

ABSTRACT

This paper is a technical report for DCASE Challenge 2022 Task 2. Our submitted model consists of a self-supervised CNN model that predicts attribute information. We have ensembled three models but have not changed the architecture, and have achieved performance improvement only by changing the data augmentation, training method, and anomaly detection method. Self-supervised learning with label information has been a powerful method in previous anomaly detection competitions, and we argue that it is equally powerful in this competition.

***Index Terms*—** anomalous sound detection, self supervised learning, domain generalization, DCASE Challenge 2022 Task2

1. INTRODUCTION

The main objective of DCASE Challenge 2022 Task 2[1] is to develop an anomalous sound detection system that supports domain generalization. The major difference from the anomalous sound detection task of Challenge 2021 Task 2, which also assumed domain shifting, is the lack of domain information during inference.

This means that participants cannot develop different models for each domain, so they need to develop a "domain generalized" model. In the past Challenge 2021 Task 2[2], a self-supervised approach was widely used to develop a feature extractor that identifies machine attributes. In this method, anomaly detection is performed by Outlier Exposure (OE)[3] as in the MobileNetV2-based baseline[1]. Another approach was to apply classical anomaly detection algorithms such as LOF and GMM to the embedding vectors of the obtained feature extractors[4][5].

Our approach is inspired by these self-supervised learning-based anomaly detection methods. In addition, we aim to improve performance by ensembling models with different attribute label classification, data augmentation, and anomaly detector.

2. ANOMALOUS SOUND DETECTION SYSTEMS

2.1. Section and attribute classification

Similar to MobileNetV2-based baseline, our method trains a classifier using the label information contained in each audio data. Specifically, one sample of 10 seconds of audio is converted to a mel-spectrogram, and a 2D Convolution-based feature extractor classifies sections 0-5. The difference from the baseline is that attribute information is trained in addition to the section information in a multitasking manner; when attribute information is used as a label, all attribute information in the training dataset is considered as a categorical label. That is, all combinations of d1p and d1v information in the attribute csv files provided by the organizer are used (e.g. f-n_A, f-n_D, loc_B...). The d1v information is used for each machine. In this way, 12-39 classes of attribute labels can be created for each machine type. For these 2 types of labels, the following equation is the loss function using Adacos[6] and Focal Loss[7].

$$L = \mathrm{FL}(\mathrm{Adacos}_{\mathrm{sec}}(\mathrm{z})) + \mathrm{FL}(\mathrm{Adacos}_{\mathrm{att}}(\mathrm{z})) \tag{1}$$

where z is the output from the feature extractor, $\mathrm{FL}(\cdot)$ is the Focal loss function, and $\mathrm{Adacos}_{\mathrm{sec/att}}(\cdot)$ are AdaCos functions with different parameters for each with section and attribute labels. EfficientNet-B1 of PyTorch Image Models, pre-trained on ImageNet, was used as the feature extractor.

2.2. Model architectures

Our submitted model is an ensemble of three anomaly detection models, each with different Data Augmentation, training and anomaly detection methods. The methods applied to each model are listed in Table 1.

2.2.1. CNN-GLOBAL

In CNN-GLOBAL, the feature extractor was trained by considering one audio data as one sample. We also employed Mixup[8] and SpecAugment[9] time masking and frequency masking as data extensions. We also adopted Gaussian-AD[10] to calculate anomaly scores. In computing the anomaly score of Gaussian-AD, the intermediate output of the trained feature extractor is used as the feature value for anomaly detection based on the Mahalanobis distance. The architecture of the feature extractor is shown in Table 2. As shown in Table 2, EfficientNet-B1 consists of 9 stages. In Gaussian-AD, GlobalAveragePooling is applied to the output of each Stage, and the vector of the Channels dimension is used as the feature (e.g. 32+16+24+40+80+112+192+320+1280=2096). It then computes the parameters of the multivariate normal distribution (MVG) using all samples in the training dataset. During in-

Table 1: Methods applied to each model

Model	Augmentation			Training approach		Anomaly detector	
	Mixup	SpecAugment	SevenBandParametricEQ	Sliding window	Two-step training	Gaussian-AD	OE
CNN-GLOBAL	✓	✓				✓	
CNN-LOCAL	✓	frequency mask only		✓		✓	
CNN-OE	✓	✓	✓		✓		✓

ference, the Mahalanobis distance from the MVG is calculated and used as the anomaly score.

Table 2: EfficientNet-B1 architecture

Stage	Operator	Resolution	Channels	Layers
Input	-	224, 313	3	-
1	Conv	112, 157	32	1
2	MBConv	112, 157	16	2
3	MBConv	56, 79	24	3
4	MBConv	28, 40	40	3
5	MBConv	14, 20	80	4
6	MBConv	14, 20	112	4
7	MBConv	7, 10	192	5
8	MBConv	7, 10	320	2
9	Conv	7, 10	1280	1

2.2.2. CNN-LOCAL

CNN-LOCAL applies an anomaly detection model with a time sliding window as in MobileNetV2-based baseline. Most of the model details are the same as in CNN-GLOBAL, but the model is randomly cropped from the mel-spectrogram with a time window size T when training the model. In addition, when computing the MVG parameters, the model is applied with a hop size H, and the MVG of the training dataset are computed. During inference, the model is applied with a time window size T to compute the anomaly score for each time frame t, and the average anomaly score for all time frames is used as the final anomaly score. For SpecAugment, only the frequency dimension was masked to prevent most of the mel-spectrogram from being masked.

2.2.3. CNN-OE

CNN-OE adopts Outlier Exposure (OE) as the anomaly detector, but most of the methods are the same as CNN-GLOBAL. In this model, the weights learned in CNN-GLOBAL are fine-tuned (two-step training). SevenBandParametricEQ from Audiomentations[11] is used as the data enhancement method, which applies filters to seven different frequency bands and adjusts the volume randomly. In CNN-OE training, SevenBandParametricEQ is randomly applied at sampling time (p=0.5) and labeled as a new section. For example, if it is applied to the section 0 sample, it is learned as the 6th class label, resulting in a total of 12 section labels. However, the Attribute label is not changed and the original label is used.

3. EVALUATION EXPERIMENT

The dataset used was the DCASE2022 Task2 Dataset. This dataset consists of MIMII DG [12]and ToyADMOS2[13] and contains sounds produced by 5 industrial products and 2 toys. Each machine has 6 different domain-shifting attributes called sections and a label indicating the machine's state and noise type called Attributes.

The input data is converted to a logmel spectrogram with window size 1024, hop size 512, and 224 melbins. In developing MVG using the training data, MVG for each section are developed using samples from each section. In other words, MVG are developed for a total of 6 sections, and anomaly scores are calculated using the MVG corresponding to each section during inference. For CNN-LOCAL with a sliding window, T=64 frames are randomly cropped from the logmel spectrogram timeframe.

When calculating the parameters of MVG, the model is applied with a hop width of H=8. During inference, the model is applied to all time frames and the anomaly score for each time frame is output.

The harmonic mean AUC / pAUC scores for each model are shown in Table 3, Table 4, Table 5, Table 6, Table 7 and Table 8. Each value represents the harmonic mean score of the development dataset, where Total (Ave.) is the average score for each model. Our best represents the value with the highest score among the three models.

4. SUBMISSIONS

The submitted models are shown below. Our submitted models are the three stand-alone models shown in Table 3 and an ensemble of all models. In the ensemble, we adopt the model with the highest score for each machine shown in Our Best.

- **Narita_AIT_task2_1** : CNN-GLOBAL
- **Narita_AIT_task2_2** : CNN-LOCAL
- **Narita_AIT_task2_3** : CNN-OE
- **Narita_AIT_task2_4** : Ensemble

5. CONCLUSION

This paper presented a self-supervised CNN model trained using three different methods. The models presented focused on modifying the data augmentation, training approach, and anomaly detection methods. In addition, the ensemble achieved an AUC of 79.73% and pAUC of 70.65% on the development dataset. However, the model presented does not directly adapt to domain shifts. Therefore, further performance improvements can be expected by introducing a Few shot domain adaptation method or adopting a more domain-robust architecture.

6. REFERENCES

[1] K. Dohi, K. Imoto, N. Harada, D. Niizumi, Y. Koizumi, T. Nishida, H. Purohit, 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 arXiv e-prints: 2206.05876*, 2022.

[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 sound detection for machine condition monitoring under domain shifted conditions."

[3] D. Hendrycks, M. Mazeika, and T. Dietterich, "Deep anomaly detection with outlier exposure," *arXiv preprint arXiv:1812.04606*, 2018.

[4] K. Morita, T. Yano, and K. Tran, "Anomalous sound detection using cnn-based features by self supervised learning," DCASE2021 Challenge, Tech. Rep., July 2021.

[5] K. Wilkinghoff, "Utilizing sub-cluster adacos for anomalous sound detection under domain shifted conditions," DCASE2021 Challenge, Tech. Rep., July 2021.

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

[7] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

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

[9] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," 2019.

[10] O. Rippel, P. Mertens, and D. Merhof, "Modeling the distribution of normal data in pre-trained deep features for anomaly detection," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 6726–6733.

[11] I. Jordal, A. Tamazian, E. T. Chourdakis, C. Angonin, askskro, N. Karpov, O. Sarioglu, kvilouras, E. B. Çoban, F. Mirus, J.-Y. Lee, K. Choi, MarvinLvn, SolomidHero, and T. Alumäe, "iver56/audiomentations: v0.25.0," May 2022. [Online]. Available: https://doi.org/10.5281/zenodo.6594177

[12] K. Dohi, T. Nishida, H. Purohit, R. Tanabe, T. Endo, M. Yamamoto, Y. Nikaido, and Y. Kawaguchi, "MIMII DG: Sound dataset for malfunctioning industrial machine investigation and inspection for domain generalization task," *In arXiv e-prints: 2205.13879*, 2022.

[13] 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 *Proceedings of the 6th Detection and Classification of Acoustic Scenes and Events 2021 Workshop (DCASE2021)*, Barcelona, Spain, November 2021, pp. 1–5.

Table 3: Harmonic mean AUC (Total)

Model	ToyCar	ToyTrain	bearing	fan	gearbox	slider	valve	Total (Ave.)
CNN-Global	37.55	55.64	71.52	26.62	85.92	78.89	**84.71**	62.98
CNN-Local	**81.44**	**66.12**	**75.64**	27.02	81.63	78.31	72.53	68.96
CNN-OE	68.34	63.82	53.86	**75.35**	**86.54**	**88.32**	67.88	72.01
Our Best	81.44	66.12	75.64	75.35	86.54	88.32	84.71	79.73

Table 4: Harmonic mean pAUC (Total)

Model	ToyCar	ToyTrain	bearing	fan	gearbox	slider	valve	Total (Ave.)
CNN-Global	53.21	50.69	70.92	61.81	63.17	69.67	**75.78**	63.61
CNN-Local	61.58	**60.51**	**72.39**	61.65	61.67	68.05	60.73	63.80
CNN-OE	**62.21**	56.50	63.92	**72.01**	**74.05**	**77.59**	58.23	66.36
Our Best	62.21	60.51	72.39	72.01	74.05	77.59	75.78	70.65

Table 5: Harmonic mean AUC (Source)

Model	ToyCar	ToyTrain	bearing	fan	gearbox	slider	valve	Total (Ave.)
CNN-Global	25.43	55.40	59.94	17.76	**89.20**	94.04	**91.64**	61.91
CNN-Local	**81.21**	**75.10**	**65.90**	18.32	84.71	93.34	85.26	71.98
CNN-OE	61.04	67.36	56.15	**74.97**	88.28	**94.17**	71.31	73.33
Our Best	81.21	75.10	65.90	74.97	89.20	94.17	91.64	81.74

Table 6: Harmonic mean AUC (Target)

Model	ToyCar	ToyTrain	bearing	fan	gearbox	slider	valve	Total (Ave.)
CNN-Global	71.75	55.88	88.64	53.10	82.87	67.95	**78.75**	71.28
CNN-Local	**81.68**	50.67	**88.76**	51.48	78.77	67.44	63.11	68.84
CNN-OE	77.60	**60.62**	51.75	**75.74**	**84.87**	**83.15**	64.77	71.21
Our Best	81.68	60.62	88.76	75.74	84.87	83.15	78.75	79.08

Table 7: Harmonic mean pAUC (Source)

Model	ToyCar	ToyTrain	bearing	fan	gearbox	slider	valve	Total (Ave.)
CNN-Global	50.31	48.58	65.40	62.33	75.30	**93.73**	**83.18**	68.40
CNN-Local	**63.12**	**63.35**	**66.15**	62.12	72.76	88.25	63.71	68.50
CNN-OE	58.25	56.87	63.35	**72.07**	**80.11**	86.96	59.34	68.14
Our Best	63.12	63.35	66.15	72.07	80.11	93.73	83.18	74.53

Table 8: Harmonic mean pAUC (Target)

Model	ToyCar	ToyTrain	bearing	fan	gearbox	slider	valve	Total (Ave.)
CNN-Global	56.47	53.00	77.47	61.29	54.41	55.45	**69.60**	61.10
CNN-Local	60.11	50.05	**79.93**	61.19	53.51	55.38	58.01	59.74
CNN-OE	**66.74**	**56.15**	64.49	**71.94**	**68.84**	**70.04**	57.15	65.05
Our Best	66.74	56.15	79.93	71.94	68.84	70.04	69.60	69.03