ANOMALOUS SOUND DETECTION BY END-TO-END TRAINING OF OUTLIER EXPOSURE AND NORMALIZING FLOW WITH DOMAIN GENERALIZATION TECHNIQUES

Technical Report

Takuya Fujimura[1], *Ibuki Kuroyanagi*[1], *Tomoki Hayashi*[1,2], *Tomoki Toda*[1]

[1] Nagoya University, School of Informatics, Nagoya, Japan
[2] Human Dataware Lab. Co., Ltd., Nagoya, Japan

ABSTRACT

In this report, we propose an anomalous sound detection (ASD) method for DCASE 2023 Challenge Task 2. Our proposed method is an extension of the serial approach using an outlier exposure-based feature extractor and an inlier modeling-based anomalous detector. We newly employ the normalizing flow as the inlier model and jointly optimize it with the feature extractor in an end-to-end manner. Furthermore, in order to deal with the domain shift, we use some domain generalization techniques, such as the domain-invariant latent space modeling in the normalizing flow and mixup to generate the pseudo-target domain data. The anomaly scores can be calculated directly using the normalizing flow or additionally using other inlier models separately trained with the optimized feature embeddings. Our final system is made by the ensemble and achieves 69.78 % in the harmonic mean of the area under the curve (AUC) and partial AUC ($p = 0.1$) over all machine types and domains on the development set.

Index Terms— anomalous sound detection, outlier exposure, normalizing flow, domain generalization, mixup

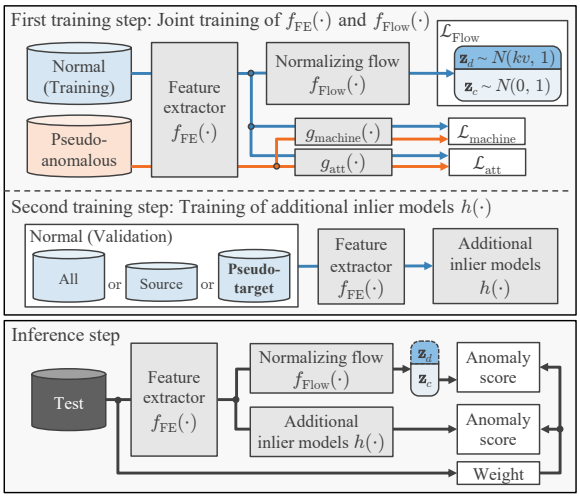

Figure 1: Overview of the proposed method

1. INTRODUCTION

This report provides the description of our submitted systems for the DCASE 2023 Challenge Task 2 [1]. This task focuses on anomalous sound detection (ASD) which aims to detect anomalous behavior of the factory machine from its sound recordings. The main difference in this task between DCASE 2022 [2] and 2023 is the limitation of the development dataset. Specifically, the variation of the normal training data expressed as the "section" is limited. Also, the machine types are completely different for development and evaluation, and the dataset for hyperparameter tuning is not provided. There is a need to create systems that achieve high performance within these limitations.

As our system, we developed the extension of the conventional serial approach Serial-OE [3], [4]. Serial-OE has two stages for training: training of feature extractor and inlier models. First, the feature extractor is trained by the classification of normal and pseudo-anomalous data, and the classification of the difference in the normal data using sections. Next, the inlier models are trained on the features of the normal data obtained by the feature extractor created in the first stage. During inference, we can identify the anomalous sound because the feature deviates from the distribution of those of the normal sound. We extend this Serial-OE by employing the normalizing flow as an inlier model and jointly optimizing it with the feature extractor. For the training of the feature extractor, we utilize attribute information instead of section IDs since they are not provided in DCASE 2023 Challenge. The anomaly score is calculated by directly using the log-likelihood of the normalizing flow or using other inlier models as in the Serial-OE. We expect that the joint optimization will result in constructing a better feature embedding space. Furthermore, we adopt domain generalization techniques for inlier models to improve performance. For the normalizing flow, we use the domain-invariant latent space modeling by discarding the latent variables which have the domain-dependent information [5]. For the training of the other inlier models, we generate the pseudo-target domain training data by mixup the source and target domain data. In addition to the domain generalization techniques, we use a simple weighting of the anomaly score based on the power of the signal. This technique enables us to properly handle the event absence period of the recordings.

We conduct an experimental evaluation using the training and test data of DCASE 2023 Task 2 Challenge development set [6], [7]. The experimental results show that all of our systems outperform the official baselines in the official metric which is the harmonic mean of the area under the curve (AUC) and partial AUC ($p = 0.1$) overall machine types and domains. Especially, the final submitted system created by an ensemble of several systems achieves 69.78% in the official metric while that of the baseline is 55.02%.

2. SERIAL-OE

As the base of our system, we used Serial-OE [4] which uses the outlier exposure (OE) [8]-based feature extractor and inlier modeling serially. Serial-OE has two stages for the training. First, the feature extractor is trained with the multi-task loss function $\mathcal{L}_{\mathrm{SerialOE}}$.

$$\mathcal{L}_{\mathrm{SerialOE}} = \mathcal{L}_{\mathrm{machine}} + \lambda_{\mathrm{section}}\mathcal{L}_{\mathrm{section}}, \tag{1}$$

where $\lambda_{\mathrm{section}}$ is a hyperparameter. The first term $\mathcal{L}_{\mathrm{machine}}$ is a loss function of the basic OE and is calculated as follows:

$$\begin{aligned}\mathcal{L}_{\mathrm{machine}} = -\frac{1}{M}\sum_{i=1}^{M}\{t_i \log\left(\sigma\left(g_{\mathrm{machine}}\left(f_{\mathrm{FE}}\left(\mathbf{x}_i\right)\right)\right)\right) \\ +(1-t_i)\log\left(1-\sigma\left(g_{\mathrm{machine}}\left(f_{\mathrm{FE}}\left(\mathbf{x}_i\right)\right)\right)\right)\},\end{aligned} \tag{2}$$

where $\mathbf{x}_i$ $(i = 1, 2, ..., M)$ is audio input, t_i $(i = 1, 2, ..., M)$ represents the machine type, M is a mini-batch size, f_{FE} is a feature extractor, g_{machine} is a function that transforms the high-dimensional embeddings to the scalar, and σ is a sigmoid function. t_i is 1 for the target machine type and 0 for the other machine types (i.e., pseudo-anomalous data). In [4], g_{machine} is an affine transformation of the norm of the embeddings. In addition to $\mathcal{L}_{\mathrm{machine}}$, Serial-OE uses the following $\mathcal{L}_{\mathrm{section}}$ to construct the detailed feature space for normal sounds:

$$\begin{aligned}\mathcal{L}_{\mathrm{section}} = -\frac{1}{K\sum_{i=1}^{M} t_i}\sum_{i=1}^{M}\sum_{k=1}^{K} t_i\{y_{i,k}\log\left(\sigma\left(g_{\mathrm{section}}\left(f_{\mathrm{FE}}(\mathbf{x}_i)\right)\right)\right) \\ +(1-y_{i,k})\log\left(1-\sigma(g_{\mathrm{section}}\left(f_{\mathrm{FE}}(\mathbf{x}_i)\right))\right)\},\end{aligned} \tag{3}$$

where each machine type has K section IDs and normal data $\mathbf{x}_i$ belongs to one of them. $y_{i,k}$ $(i = 1, 2, ..., N, k = 1, 2, ..., K)$ is the one-hot vector for the section ID where $y_{i,k}$ is 1 for the k th element and 0 for the other elements when the section ID is k. g_{section} is a linear transformation. This multi-task training constructs a better embedding space in which both anomalous sounds that are not similar to normal sounds and anomalous sounds that are similar to normal sounds are distinguished from normal sounds.

Next, we train inlier models such as Gaussian mixture models (GMM) [9], [10] and k-nearest neighbor algorithm (KNN) [11] with normal data. These inlier models are used as the anomaly detector h and the anomaly score a_i is calculated as follows:

$$a_i = \mathcal{A}\left(h\left(f_{\mathrm{FE}}\left(\mathcal{X}_i\right)\right)\right), \tag{4}$$

where $\mathcal{X}_i$ is a set of S segments that divide $\mathbf{x}_i$ into T seconds with overlap and $\mathcal{A}$ is an aggregator of the anomaly scores such as average operation. Serial-OE has achieved high performance by modeling normal sound distribution in the suitable feature space for ASD.

3. PROPOSED METHOD

Based on the success of the Serial-OE, we extend it by using a normalizing flow as a neural inlier model and jointly optimizing it with a feature extractor. Figure 1 shows an overview. In the first step, we jointly optimize the networks with the following $\mathcal{L}_{\mathrm{OEFlow}}$.

$$\mathcal{L}_{\mathrm{OEFlow}} = \mathcal{L}_{\mathrm{SerialOE}} + \lambda_{\mathrm{Flow}}\mathcal{L}_{\mathrm{Flow}}, \tag{5}$$

$$\mathcal{L}_{\mathrm{Flow}} = -\frac{1}{M}\sum_{i=1}^{M}\log p_X(\mathbf{x}_i^{\mathrm{FE}}), \tag{6}$$

$$\log p_X(\mathbf{x}_i^{\mathrm{FE}}) = \log p_Z(f_{\mathrm{Flow}}(\mathbf{x}_i^{\mathrm{FE}})) + \log\left|\det\left(\frac{\partial f_{\mathrm{Flow}}(\mathbf{x}_i^{\mathrm{FE}})}{\partial \mathbf{x}_i^{\mathrm{FE}}}\right)\right|, \tag{7}$$

where $\mathbf{x}_i^{\mathrm{FE}} = f_{\mathrm{FE}}(\mathbf{x}_i)$, p_X is an input data distribution, and p_Z is a standard normal distribution $N(0, 1)$. f_{Flow} is a composition of multiple invertible transformations and we employ a FastFlow [12] as f_{Flow}. We expect that the end-to-end training with $\mathcal{L}_{\mathrm{OEFlow}}$ enables the feature extractor to extract the more suitable feature for normalizing flow and make a better embedding space where the anomalous data is distinguished from the normal data. Also, in the DCASE2023 Challenge, each machine type has only one section. Therefore, we use the following $\mathcal{L}_{\mathrm{att}}$ instead of $\mathcal{L}_{\mathrm{section}}$.

$$\begin{aligned}\mathcal{L}_{\mathrm{att}} = -\frac{1}{K^{\mathrm{att}}M}\sum_{i=1}^{M}\sum_{k=1}^{K^{\mathrm{att}}}\{y_{i,k}^{\mathrm{att}}\log\left(\sigma\left(g_{\mathrm{att}}\left(f_{\mathrm{FE}}(\mathbf{x}_i)\right)\right)\right) \\ +(1-y_{i,k}^{\mathrm{att}})\log\left(1-\sigma(g_{\mathrm{att}}\left(f_{\mathrm{FE}}(\mathbf{x}_i)\right))\right)\},\end{aligned} \tag{8}$$

where g_{att} is a linear transformation. The attribute information of the target machine has K^{att} values and $y_{i,k}^{\mathrm{att}}$ is a one-hot vector for it. For the pseudo-anomalous data, $y_{i,k}^{\mathrm{att}}$ is set to 0.

In the second step, we train the additional inlier models such as GMM and KNN as in the Serial-OE. In the inference, the anomaly score is calculated as in Eq. 4 directly using the negative log-likelihood of the normalizing flow as $h(\mathbf{x}_i^{\mathrm{FE}}) = -\log p_Z(f_{\mathrm{Flow}}(\mathbf{x}_i^{\mathrm{FE}}))$ or using additional inlier models.

3.1. Domain generalization technique

One of the main points of DCASE 2023 Challenge Task 2 is domain generalization. Therefore, we introduce two domain generalization techniques into the training of the inlier models.

Domain-invariant latent space modeling: To alleviate the domain shift in the ASD with the normalizing flow, we use domain-invariant latent space modeling [5]. In this framework, some latent variables $\mathbf{z}_d$ are constrained to follow $N(kv, 1)$ while the others $\mathbf{z}_c$ are constrained to follow $N(0, 1)$, where k is a hyperparameter and v represents the physical parameter which causes domain shift (e.g., operation velocity). Creating $\mathbf{z}_d$ makes $\mathbf{z}_c$ invariant to the physical parameter and the domain shift-invariant anomaly scores are calculated as $a = -\log p_Z(\mathbf{z}_c)$. We introduce this domain generalization technique by splitting channels of the latent variables of the FastFlow into $\mathbf{z}_c$ and $\mathbf{z}_d$.

Generating pseudo-target domain data by mixup: To alleviate the domain shift in ASD with additional inlier models, we use the pseudo-target domain data for the training of the second step. The pseudo-target domain data is generated by mixup the source domain data with the target domain data. Utilizing pseudo-target domain data for the training reduces false positives due to the domain shift.

3.2. Weighting anomaly score

To further improve the performance of our systems, we handle the event absence problems. In the DCASE 2023 Challenge Task 2, the recordings of some machine types (e.g., ToyCar and ToyTrain) have an event-absence period, especially at the beginning and end of the audio. To prevent false positives in the event-absence period, we adopt the weighted mean as the aggregator $\mathcal{A}$. The weight is calculated based on the power of the signals and it is pre-processed with a high-pass filter to remove the effect of the noise. For the ASD with the normalizing flow, we directly apply weighting to the anomaly score map provided by FastFlow which reflects the shape of the input feature (i.e., it has a time and frequency axis).

Table 1: Summary of our systems.

Systems	loss function	Domain-invariant modeling [5]
OE	$\mathcal{L}_{\mathrm{machine}}$	Not applicable
OE-Flow	$\mathcal{L}_{\mathrm{machine}} + \lambda_{\mathrm{Flow}}\mathcal{L}_{\mathrm{Flow}}$	Not applied
OE-DFlow	$\mathcal{L}_{\mathrm{machine}} + \lambda_{\mathrm{Flow}}\mathcal{L}_{\mathrm{Flow}}$	Applied
SerialOE-Flow	$\mathcal{L}_{\mathrm{SerialOE}} + \lambda_{\mathrm{Flow}}\mathcal{L}_{\mathrm{Flow}}$	Not applied
SerialOE-DFlow	$\mathcal{L}_{\mathrm{SerialOE}} + \lambda_{\mathrm{Flow}}\mathcal{L}_{\mathrm{Flow}}$	Applied

4. EXPERIMENTAL EVALUATIONS

4.1. Systems

We developed several types of the proposed method as shown in Table 1. For the machines that have multiple attribute parameters, multiple systems are developed for OE-DFlow and SerialOE-DFlow using each parameter separately. The separately developed systems using a different attribute item were ensembled for each inlier model and its hyperparameter.

SerialOE-Flow and SerialOE-DFlow did not use additional inlier models because there are not enough samples to model the feature embeddings space for each parameter in the target domain. The other systems use additional inlier models and ensemble the twenty inlier models with several hyperparameters including the normalizing flow based on the evaluation results of the development set. We used the official Autoencoder-based baseline systems including the selective Mahalanobis mode [13] as the baseline.

4.2. Experimental setups

We conducted an experimental evaluation using the DCASE 2023 Task 2 Challenge development sets (ToyADMOS2 [6], MIMII DG [7]). The development sets included seven machine types: bearing, fan, gearbox, valve, slider, ToyCar, and ToyTrain. The training data had 1,000 samples of normal data for each machine type, of which 990 samples are in the source domain and ten samples are in the target domain. The test data had 50 samples of normal and anomalous data for each machine type and each domain. Each recording was a 10 or 12-second single channel segment sampled at 16 kHz. For the training, we used 85 % of the source domain data and six samples of the target domain data. The remaining samples were treated as the validation set. For the pseudo-anomalous data, we used the training data of the non-target machine type and additional training datasets of the DCASE 2023 Task 2 Challenge.

The amplitude of the audio input sequence was standardized to have a mean of 0.0 and a variance of 1.0. The audio input sequence was extracted as Mel-spectrogram with a window size of 128 ms, a hop size of 16 ms, and 224 Mel-spaced frequency bins in the range of 50 Hz to 7800 Hz in 5.0 seconds. The feature was passed to the f_{FE} of ResNet-18 [14]. We used a linear transformation for g_{machine}. For the OE in Table 1, we trained the feature extractor f_{FE} for 150 epochs with an OneCycleLR [15] of learning rate 0.001. For the other systems, we trained the networks for 250 epochs with a fixed learning rate 0.0001. The optimization algorithm was AdamW [16] and the batch size was 128. When creating mini-batches, we used a batch sampler so that the value of t is 1:1. $\lambda_{\mathrm{section}}$ in Eq. 1 was set to 5 and λ_{Flow} in Eq. 5 was set to 10^{-7}. We used a mixup in the training of the feature extractor. This mixup aimed to obtain intermediate features between normal and pseudo-anomalous data or different parameters of normal data while mixup in Sec. 3.1 aimed to generate pseudo-target domain data for the training of the additional inlier models. The normalizing flow was trained with only samples that do not contain pseudo-anomalous components. For the domain-invariant latent space modeling in the normalizing flow, we also used categorical labels by expressing it as an integer value v. When using the data generated by mixup of normal data with different parameters, we used the parameter of the dominant one for v. The hyperparameter k was set to 5 divided by the minimum distance of the parameters. We used eight of the 512 channels of the latent variables as $\mathbf{z}_d$.

During inference, we divided the original clips into S segments with 75% overlapping. As the additional inlier model, we used GMM and KNN, and the hyperparameter was the number of components for GMM or the number of neighbors for KNN, where it was one of $\{1, 2, 4, 16 \text{ and } 32\}$. The GMM used the negative log-likelihood as the anomaly score, while the KNN used the distance to the nearest points with Mahalanobis distance. For the training data in the second step, we had three options: source domain only, pseudo-target data, and original data from both domains, where this selection was treated as a hyperparameter. The aggregator $\mathcal{A}$ was weighted mean described in Sec. 3.2 for ToyCar and ToyTrain and mean for the others. In the evaluation set, we used a weighted mean for ToyDrone, ToyNscale, ToyTank, and Vacuum. The passband and stopband edge frequencies of the high-pass filter used in calculating the weight were set to the 4000 kHz and 3500 kHz.

4.3. Experimental results

Tables 2, 3, and 4 show the performance of the source domain, target domain, and both domains, respectively. Each table summarizes the evaluation results of both the original system in Table 1 and four submitted systems that are made by ensembling the several systems.

As a result, all of our final submitted systems and original single systems outperformed the performance of the official baselines. We can also see that joint optimization methods such as OE-Flow and OE-DFlow outperformed the OE. Considering that the normalizing flow has not been used as the anomaly detector h in the OE-DFlow, it can be assumed that a better feature embedding space has been made through the joint optimization of the feature extractor and the normalizing flow. We can confirm that SerialOE-DFlow has also achieved competitive results except for the specific case (valve in the target domain). In our experiments, we confirm the performance improvement by domain-invariant latent space modeling in the source domain. A detailed analysis is included in future work.

5. CONCLUSION

We proposed ASD methods for the DCASE 2023 Challenge Task 2. The proposed method was an extension of the Serial-OE and we employed a normalizing flow as the inlier model. The inlier model and feature extractor networks were jointly optimized with a multi-task loss function. The anomaly scores were calculated using the normalizing flow and additional inlier models. In addition to proposing a new ASD method, we adopted some techniques to handle the domain shift and event absence problems. In the evaluation of the development set, all of our systems outperform the baselines and they achieved absolute improvements of up to 14% in the official score. Also, the results suggested that joint optimization of the feature extractor with the normalizing flow could construct a better feature space for ASD. The future work includes a detailed analysis of our framework of joint optimization and its improvement.

6. ACKNOWLEDGMENT

This paper was partly supported by a project, JPNP20006, commissioned by NEDO, and JSPS KAKENHI Grant Number JP20H00102.

Table 2: Evaluation results in the source domain. The values represent AUC [%] for the source domain. The value in the column "hmean" represents the harmonic mean of AUC over all machines.

System	Inlier models or systems used for ensemble	bearing	fan	gearbox	slider	valve	ToyCar	ToyTrain	hmean
Baseline (MSE)		65.92	80.19	60.31	70.31	55.35	70.10	57.93	64.79
Baseline (Mahalanobis)		65.16	87.10	71.88	84.02	56.31	74.53	55.98	68.84
OE	GMM and KNN	72.48	71.12	**86.92**	98.48	93.76	55.84	50.12	71.40
OE-Flow	GMM, KNN, and Normalizing flow	**75.64**	73.76	85.08	**99.84**	93.20	54.68	54.44	72.97
OE-DFlow	GMM and KNN	73.04	79.28	81.96	99.52	98.24	60.04	56.92	75.30
SerialOE-Flow	Normalizing flow	67.64	51.40	68.40	98.12	**99.04**	57.12	61.40	68.00
SerialOE-DFlow	Normalizing flow	74.32	70.88	68.44	98.96	94.64	63.28	**69.84**	75.29
Submitted system1	Our all methods	73.16	72.76	83.12	99.80	97.64	59.28	67.84	76.66
Submitted system2	OE-DFlow and SerialOE-DFlow	73.68	**85.64**	77.16	99.52	96.92	**64.20**	69.08	**78.98**
Submitted system3	OE, OE-Flow, and OE-DFlow	74.12	75.72	87.20	99.72	96.32	59.88	53.16	74.40
Submitted system4	Our methods except for SerialOE-Flow	74.44	78.04	83.72	99.72	97.20	62.36	68.88	78.60

Table 3: Evaluation results in the target domain. The values represent AUC [%] for the target domain. The value in the column "hmean" represents the harmonic mean of AUC over all machines.

System	Inlier models or systems used for ensemble	bearing	fan	gearbox	slider	valve	ToyCar	ToyTrain	hmean
Baseline (MSE)		55.75	36.18	60.69	48.77	50.69	46.89	57.02	49.59
Baseline (Mahalanobis)		55.28	45.98	70.78	73.29	51.40	43.42	42.45	52.37
OE	GMM and KNN	45.60	67.84	71.88	90.08	**100.00**	53.36	55.28	64.51
OE-Flow	GMM, KNN, and Normalizing flow	59.32	57.44	78.60	**94.72**	99.60	59.00	54.72	68.09
OE-DFlow	GMM and KNN	**61.32**	64.76	76.80	89.88	**100.00**	61.84	53.96	69.58
SerialOE-Flow	Normalizing flow	58.48	64.44	76.56	93.00	18.60	58.44	54.44	48.05
SerialOE-DFlow	Normalizing flow	60.36	**78.48**	71.84	88.40	11.92	60.52	**56.24**	40.52
Submitted system1	Our all methods	58.04	71.88	**82.44**	92.48	97.60	62.12	54.48	**70.86**
Submitted system2	OE-DFlow and SerialOE-DFlow	60.80	66.52	77.80	90.04	73.88	**64.44**	55.36	68.25
Submitted system3	OE, OE-Flow, and OE-DFlow	57.48	70.36	77.56	91.96	**100.00**	59.60	54.80	69.72
Submitted system4	Our methods except for SerialOE-Flow	57.40	70.96	79.44	91.88	98.68	62.48	55.24	70.54

Table 4: Evaluation results. The values represent the harmonic mean of AUC and pAUC over all domains. "official" represents the official score which is calculated by harmonic mean of AUC and pAUC over all machine types and domains.

System	Inlier models or systems used for ensemble	bearing	fan	gearbox	slider	valve	ToyCar	ToyTrain	official
Baseline (MSE)		56.67	52.59	57.86	57.18	52.33	54.89	54.16	55.02
Baseline (Mahalanobis)		56.71	59.90	64.60	68.46	52.83	52.83	48.23	56.91
OE	GMM and KNN	54.74	67.78	69.94	87.24	93.47	52.98	51.03	64.84
OE-Flow	GMM, KNN, and Normalizing flow	63.52	64.49	74.20	**93.35**	92.40	55.02	52.52	67.65
OE-DFlow	GMM and KNN	62.02	71.66	72.88	86.53	**97.23**	58.08	53.03	68.82
SerialOE-Flow	Normalizing flow	60.95	54.84	67.49	88.92	36.25	54.56	54.50	56.01
SerialOE-DFlow	Normalizing flow	**63.24**	67.97	66.33	83.39	26.22	58.88	**57.11**	53.76
Submitted system1	Our all methods	61.64	71.64	75.53	90.22	90.49	57.28	55.89	69.37
Submitted system2	OE-DFlow and SerialOE-DFlow	61.97	74.19	71.41	86.49	76.31	**60.11**	56.68	68.25
Submitted system3	OE, OE-Flow, and OE-DFlow	61.05	73.19	**74.51**	89.86	95.78	56.70	51.98	68.69
Submitted system4	Our methods except for SerialOE-Flow	61.05	**74.22**	74.10	89.63	91.15	58.36	56.48	**69.78**

7. REFERENCES

[1] K. Dohi, K. Imoto, N. Harada, D. Niizumi, Y. Koizumi, T. Nishida, H. Purohit, R. Tanabe, T. Endo, and Y. Kawaguchi, "Description and discussion on dcase 2023 challenge task 2: First-shot unsupervised anomalous sound detection for machine condition monitoring," *In arXiv e-prints: 2305.07828*, 2023.

[2] 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.

[3] I. Kuroyanagi, T. Hayashi, K. Takeda, and T. Toda, "Two-stage anomalous sound detection systems using domain generalization and specialization techniques," DCASE2022 Challenge, Tech. Rep., 2022.

[4] I. Kuroyanagi, T. Hayashi, K. Takeda, and T. Toda, "Improvement of serial approach to anomalous sound detection by incorporating two binary cross-entropies for outlier exposure," in *Proc. European Signal Processing Conference*, 2022, pp. 294–298.

[5] K. Dohi, T. Endo, and Y. Kawaguchi, "Disentangling physical parameters for anomalous sound detection under domain shifts," in *Proc. European Signal Processing Conference*, 2022, pp. 279–283.

[6] N. Harada, D. Niizumi, D. Takeuchi, Y. Ohishi, M. Yasuda, and S. Saito, "ToyADMOS2: Another dataset of miniature-machine operating sounds for anomalous sound detection under domain shift conditions," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*, 2021, pp. 1–5.

[7] 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 ggeneralization task," in *Proc. Detection and Classification of Acoustic Scenes and Events 2022 Workshop*, 2022.

[8] D. Hendrycks, M. Mazeika, and T. Dietterich, "Deep anomaly detection with outlier exposure," *Proceedings of the International Conference on Learning Representations*, 2019.

[9] D. W. Scott, "Outlier Detection and Clustering by Partial Mixture Modeling," in *COMPSTAT 2004 — Proceedings in Computational Statistics*, 2004, pp. 453–464.

[10] W. Liu, D. Cui, Z. Peng, and J. Zhong, "Outlier Detection Algorithm Based on Gaussian Mixture Model," in *Proc. International Conference on Power, Intelligent Computing and System*, 2019, pp. 488–492.

[11] S. Ramaswamy, R. Rastogi, and K. Shim, "Efficient Algorithms for Mining Outliers from Large Data Sets," *SIGMOD Rec.*, vol. 29, no. 2, pp. 427–438, 2000.

[12] J. Yu, Y. Zheng, X. Wang, W. Li, Y. Wu, R. Zhao, and L. Wu, "Fastflow: Unsupervised anomaly detection and localization via 2d normalizing flows," *arXiv preprint arXiv:2111.07677*, 2021.

[13] N. Harada, D. Niizumi, D. Takeuchi, Y. Ohishi, and M. Yasuda, "First-shot anomaly detection for machine condition monitoring: A domain generalization baseline," *In arXiv e-prints: 2303.00455*, 2023.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015.

[15] L. N. Smith and N. Topin, "Super-convergence: very fast training of neural networks using large learning rates," in *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, vol. 11006, 2019, pp. 369–386.

[16] I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization," in *Proc. International Conference on Learning Representations*, 2019.