机器状态监测的首样本异常声音检测：域泛化基线

Noboru Harada、Daisuke Niizumi、Yasunori Ohishi、Daiki Takeuchi 和 Masahiro Yasuda*NTT Corporation*日本 电子邮箱：noboru.harada.pv@hco.ntt.co.jp

arXiv:2303.00455v1 [eess.AS] 1 Mar 2023

摘要—**本文提供了一个符合首样本要求的机器状态监测无监督异常声音检测（ASD）基线系统。首样本 ASD 不允许系统根据使用真实标签计算的性能指标进行依赖机器类型的超参数调优或工具集成。为了展示首样本 ASD 的基准性能，本文提出了一种异常声音检测系统，该系统适用于声学场景与事件检测和分类（DCASE）2022年挑战赛任务 2 中的域泛化任务“应用域泛化技术进行机器状态监测的无监督异常声音检测”，同时遵循 DCASE 2023 挑战赛任务 2（DCASE2023T2）提出的首样本要求。本文实现了一个简单的、基于自编码器并结合选择性马氏距离的基线系统。通过性能评估，为即将举行的 DCASE2023T2 设定目标基准。该基线系统的源代码将在 GitHub 上提供。**

索引词——**首次提交异常声音检测、域泛化、机器状态监测**

一、引言

本文为符合首次提交要求的机器状态监测无监督异常声音检测（ASD）提供了一个基线系统。

利用深度学习技术进行机器状态监测以实现预测性维护，是工业 2.0/4.0 中一项至关重要的应用。在声学场景与事件检测和分类（DCASE）挑战赛中，设有无监督异常声音检测任务[1]，[2]，[3]。然而，大多数获胜系统都采用了针对挑战赛任务设置的专用技术。在该任务设置中，训练数据包含许多来自相似但不同机器实例的声音样本，并将其作为不同区段提供。一些系统将这些不同区段的数据用作伪异常数据样本。此外，大多数获胜系统都依赖于依机器类型而定的超参数调优，以及基于给定异常样本性能观察结果的工具集成，以进行性能评估。这些解决方案并不总是适用于工业界的实际应用场景。

由于 DCASE 2022 挑战赛任务 2：应用于机器状态监测的无监督异常声音检测任务规则中未规定任何限制域泛化技术（DCASE2022T2），排名前五的获胜系统 [4]，[5]，[6]，[7]，[8] 均采用了针对每种机器类型使用依赖机器类型的超参数调优和工具集成的技术。这些系统根据用于性能评估的正常样本和异常样本计算出的曲线下面积等预知性能指标进行了微调。因此，异常样本中的信息已经泄露到了系统中。

鉴于上述情况，DCASE 挑战赛 2023 任务 2：面向机器状态监测的首次无监督异常声音检测（DCASE2023T2）将在异常声音检测任务中引入首样本要求。首次异常声音检测任务不允许系统参考根据真实标签计算的性能指标进行任何超参数调优，尤其是不允许使用用于评估系统的异常样本。

为展示首样本异常声音检测的基准性能，本文提出了一种适用于 DCASE2022T2[3] 域泛化任务的异常声音检测系统，同时满足 DCASE2023T2 中新引入的首样本要求。

DCASE2022T2 是一项域泛化任务。不过，也可以将其视为一项训练数据高度不平衡的训练任务。DCASE2022T2 域泛化任务要求采用高效的无监督异常检测训练方案，以应对高度不平衡的训练数据（训练时提供来自源域的 990 个正常样本和来自目标域的仅 10 个样本），且测试目标样本时不提供域标签。

本文提出了一种简单的马氏距离实现，作为即将举行的 DCASE2023T2 挑战赛的基线系统。该策略是 Liu 重庆邮电大学任务中所用方法的微小子集2 4[4]，该任务在 DCASE2022T2 中获得了第二名。所提出基线系统的源代码将发布在 GitHub[9]上。

为满足首样本要求，系统应能在工业中遇到的任何真实数据上表现出相近的良好性能。

II. 首样本异常检测任务

A. 首样本无监督异常声音检测任务

在2020年、2021年和2022年举办的DCASE异常声音检测挑战赛中，已有研究针对机器状态监测提出了几种高效技术；其中一些技术利用来自不同区段的数据作为异常样本。此外，还有一些技术依赖于依赖机器类型的超参数调优，以及基于给定异常样本性能表现的工具集成。

通常，在现实应用中，很难从不同机器实例收集到足够多样的数据样本，也很难收集到足够的异常样本。因此，这些解决方案在此类情况下并不现实。

为了鼓励挑战赛参与者提供真正实用、可用于现实应用的技术，我们在即将举行的 DCASE2023T2 [10] 中新增了首样本要求。

首样本异常声音检测的特点如下：

- 不使用来自不同区段（不同机器实例）的数据
- 不对特定机器类型进行超参数调优
- 不对特定机器类型应用工具集成

任务条件中隐含地落实了相关要求，而不是通过陈述规则来说明，以此自然地迫使解决方案符合要求。例如，每种机器类型仅提供一个区段，因此不适用任何分类技术。此外，评估数据集（用于对系统进行排名）提供的机器类型与开发数据集给定的机器类型完全不同，因此无法通过核查真实标签来进行超参数调优或集成工具调优。

B. 将域泛化视为数据不平衡任务

本文要解决的目标任务是 DCASE2022T2：域泛化任务[11], [12], [13] ，但该任务还受到首次提交要求的限制。

本研究使用了 DCASE2022T2 评估所用的额外训练数据集和评估数据集。如图 1 所示，共有七种机器类型：玩具汽车、玩具火车、风扇、齿轮箱、轴承、滑轨（滑块）和轴承。每种机器类型包含三个区段，反映机器在不同运行条件下的状态。每个区段都包含代表域偏移条件的源域和目标域。仅在训练时提供标注源域/目标域的标签。测试时不提供域标签。训练数据共有 990 个文件

源域数据，而目标域仅提供 10 条用于训练。所有文件均包含 10 秒的正常机器运行声音。未提供异常声音样本。此任务条件反映了一种应用场景：存在多种运行模式，但可用于训练的录音数量有限。测试系统时，使用了 DCASE2022T2 的评估数据集：源域和目标域各有 50 个正常文件和 50

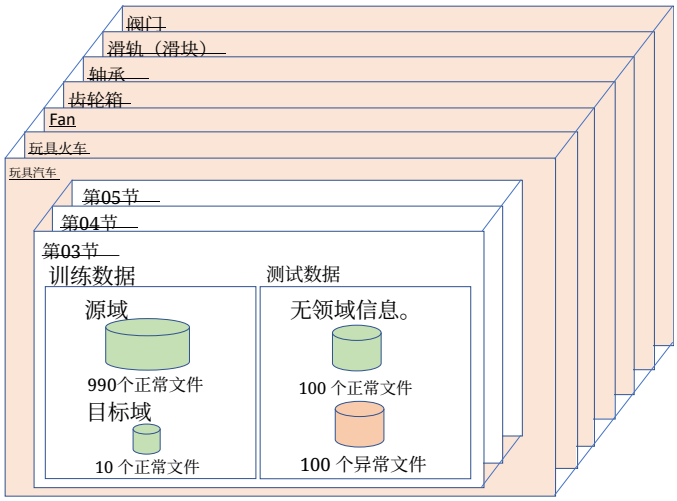


图 1：DCASE2022T2 的额外训练集和评估数据集。

个异常文件；每种机器类型的每个区段有 200 个无标签文件。

C. 性能评估指标

按照 DCASE2022T2 的排名规则，系统评估总分 Ω 基于受试者工作特征（ROC）曲线下面积（曲线下面积）和部分曲线下面积（部分曲线下面积）计算，二者的调和平均值为 $hmean$，公式如下：

$$\Omega = hmean\{AUC_{m,n,d}, pAUC_{m,n} \mid m \in \mathcal{M}, n \in \mathcal{S}(m), d \in \{source, target\}\}, \quad (1)$$

$$AUC_{m,n,d} = \frac{1}{N_d^- N_n^+} \sum_{i=1}^{N_d^-} \sum_{j=1}^{N_n^+} \mathcal{H}(A_\theta(x_j^+) - A_\theta(x_i^-)), \quad (2)$$

$$pAUC_{m,n} = \frac{1}{\lfloor pN_n^- \rfloor N_n^+} \sum_{i=1}^{\lfloor pN_n^- \rfloor} \sum_{j=1}^{N_n^+} \mathcal{H}(A_\theta(x_j^+) - A_\theta(x_i^-)). \quad (3)$$

曲线下面积是根据每个机器类型 $m \in \mathcal{M}$ 的各个域 $d \in \{source, target\}$ 上正常样本 x_i^- 和异常样本 x_j^+ 的异常分数 $A_\theta(\cdot)$ 计算得出，并在各区段 $n \in \mathcal{S}(m)$ 上取平均值，其中 m 是机器类型指示符，n 是机器类型中的区段指示符。

在每种机器类型 $m \in \mathcal{M}$ 上计算部分曲线下面积（pAUC），并对不同域中的各区段取平均值。N_d^- 是每个域中的正常样本数量。N_n^-和 N_n^+分别是机器类型 m 中区段 n 的正常样本和异常样本数量。$\mathcal{H}(\cdot)$是一个函数，当$x > 0$ 时返回 1，否则返回 0。计算部分曲线下面积时，假阳性率（FPP） p 设为 0.1。$\lfloor \cdot \rfloor$ 是向下取整函数。

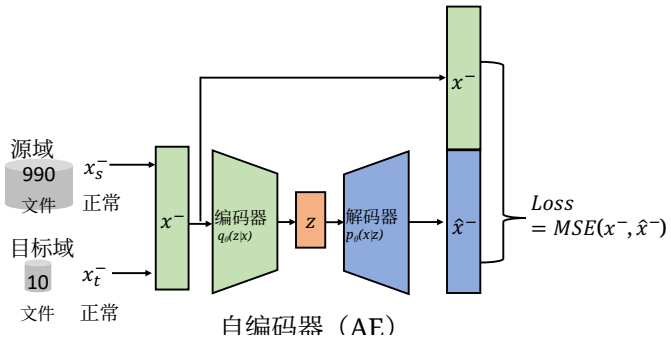


图 2：修改后的基线自编码器概览。

III. 用于首次域泛化的基线系统

DCASE2022T2 中提出了若干系统。一些获胜系统采用了基于分类的检测器。在给定的不同区段样本之间进行分类。例如，玩具汽车这一机器类型有三个区段。请注意，DCASE2022T2 中有两个基线系统：DCASE 2022 任务 2 自编码器（AE）和 MovileNetV2 [14], [15], [16]。然而，这两个基线系统均不符合首样本要求。

本文提出了一种符合首次提交要求的基线系统，为即将到来的 DCASE2023T2 提供目标性能基准。

A. 符合首次拍摄要求的基线自编码器

符合首次拍摄要求的基线自编码器基于 DCASE2022T2 基线自编码器[14]进行了少量修改，其结构如图 2 所示。

DCASE2022T2 基线自编码器使用目标机器类型中的所有区段数据进行训练，而符合首次提交要求的版本仅使用该机器类型中专属区段的目标正常样本。因此，当数据集中包含三个区段时，应分别针对每个区段训练三个模型。

输入音频样本使用短时傅里叶变换进行分析，帧长为 64 毫秒， hop 长度为 50%，随后转换为 128 个频带的对数梅尔能量。将连续五帧拼接起来，构成输入自编码器的向量，其维度为 640。

训练时，通过以下方式训练自编码器的模型参数 θ ，使输入正常样本 x^- 与其重构 $\hat{x}^-$ 之间的均方误差（MSE）最小化：

$$Loss \quad = MSE(x^-, \hat{x}^-), \tag{4}$$

$$where \quad \hat{x}^- = Dec_\theta(Enc_\theta(x^-)). \tag{5}$$

测试阶段，使用给定查询样本 x 的重构误差计算异常分数，计算方式如下：

$$Anomaly\ Score \quad A_\theta \quad = MSE(x, \hat{x}), \tag{6}$$

$$where \quad \hat{x} \quad = Dec_\theta(Enc_\theta(x)). \tag{7}$$

当异常分数超过预设阈值时，该样本会被判定为异常样本。

B. 选择性马氏距离自编码器

在 III-A 中符合首次提交要求的基线自编码器之上，可以引入马氏距离 [17] 来计算异常分数。这一策略只是 Liu 重庆邮电大学任务2 4[4]所采用方法的微小子集，而该任务是 DCASE2022T2 的第一名获胜者。

除了在训练最后一个 epoch 更新后计算源域和目标域中正常样本 x^- 与其重构 $\hat{x}^-$ 之间距离的协方差矩阵 Σ_s^{-1} 和 Σ_t^{-1} 外，其基本模型结构和训练方案均与 III-A 中所示的自编码器相同。

使用这些协方差矩阵计算异常分数 A_θ ，计算方式如下：

$$Anomaly\ Score \quad A_\theta \quad = min\{D_s(x, \hat{x}), D_t(x, \hat{x})\}, \tag{8}$$

$$where \quad D_s(\cdot) \quad = Mahalanobis(x, \hat{x}, \Sigma_s^{-1}), \tag{9}$$

$$D_t(\cdot) \quad = Mahalanobis(x, \hat{x}, \Sigma_t^{-1}). \tag{10}$$

本文将其称为选择性马氏距离自编码器。

C. 所提出基线系统的设计原则

第 III-A 节和第 III-B 节中介绍的基线模型被实现为包含两种运行模式的一份基线源代码。新的基线使用 PyTorch 实现，而之前的 DCASE2022T2 基线自编码器 [14]使用 Keras 实现。两者的网络架构等效。

我们测试了几种可能纳入的工具。例如，已对 [18], [19], [20] 中介绍的工具进行了检验。然而，与增加的复杂度（即源代码级复杂度和运行时计算复杂度）相比，从总分来看，这些工具带来的性能提升并不显著。

作为基线实现，简洁性是最重要的方面之一。因此，我们决定不纳入这些工具，而是保持其简洁性。

IV. 实验

使用第 II-B 节所述的 DCASE2022T2附加训练和评估数据集来评估所提出的基线系统的性能。采用第 II-C 节所述的性能评估指标，对符合首次拍摄要求的基线自编码器（FS-AE）和选择性马氏距离自编码器的性能得分进行评估。

短时傅里叶变换的帧长为 64 ms，帧移为帧长的 50 %，并转换为 128 个频带的对数梅尔能量。将连续五帧拼接成 640 维（128 × 5）输入系统。系统包含三个线性层，每层为 128 维，并依次使用批归一化和 ReLU 激活函数。瓶颈层为 8 维。训练轮数为 100。批量大小为 256，Adam优化器的学习率为 0.001。

表I展示了使用三个不同随机种子得到的测试结果及平均分数，并将其与 DCASE2022T2 系统的分数进行了比较。符合首次提交要求的基线自编码器和选择性马氏距离自编码器的总分应作为 DCASE2023T2 挑战赛的目标基准。

表I：曲线下面积结果 基于 DCASE 2022 挑战赛任务 2 数据集，在域泛化条件下 .

系统	指标	hmean*1	算术均值*1	玩具汽车	玩具火车	fan	齿轮箱	轴承	滑块	阀门
首次提交合规	AUC（源域）	0.6484	0.6779	0.8229	0.4943	0.6609	0.7137	0.7475	0.7563	0.5493
基线自编码器	AUC（目标域）	0.5143	0.5451	0.6571	0.5195	0.3679	0.6116	0.5639	0.5275	0.5681
（FS-AE）	pAUC（源域与目标域）	0.5335	0.5407	0.6582	0.4967	0.5114	0.5200	0.5426	0.5361	0.5195
随机种子 13711	**总分**	0.5596	0.5879							
	AUC（源域）	0.6520	0.6822	0.8509	0.4986	0.6524	0.7087	0.7493	0.7624	0.5529
	AUC（目标域）	0.5143	0.5449	0.6778	0.5251	0.3575	0.5969	0.5627	0.5293	0.5653
	pAUC（源域与目标域）	0.5367	0.5452	0.6786	0.4988	0.5175	0.5128	0.5528	0.5347	0.5211
随机种子 13591	**总分**	0.5616	0.5908							
	AUC（源域）	0.6542	0.6808	0.8253	0.5063	0.6554	0.7125	0.7468	0.7595	0.5596
	AUC（目标域）	0.5084	0.5355	0.6263	0.5283	0.3487	0.5993	0.5597	0.5222	0.5639
	pAUC（源域与目标域）	0.5329	0.5402	0.6607	0.4965	0.5118	0.5128	0.5468	0.5356	0.5170
brandom 随机种子 13267	**总分**	0.5584	0.5855							
	AUC（源域）	0.6515	0.6803	0.8331	0.4998	0.6562	0.7116	0.7479	0.7594	0.5539
	AUC（目标域）	0.5123	0.5418	0.6537	0.5243	0.3580	0.6026	0.5621	0.5264	0.5657
	pAUC（源域与目标域）	0.5344	0.5420	0.6658	0.4973	0.5136	0.5152	0.5474	0.5355	0.5192
平均值	**总分**	**0.5599**	**0.5880**							
选择性马氏距离	AUC（源域）	0.6597	0.7118	0.9528	0.4862	0.6789	0.8533	0.7216	0.7572	0.5329
AE	AUC（目标域）	0.5610	0.6082	0.8371	0.5043	0.4211	0.7801	0.6286	0.5568	0.5289
（Liu [4] 的微小子集）	pAUC（源域与目标域）	0.5660	0.5805	0.7879	0.5118	0.5221	0.6104	0.5753	0.5389	0.5170
随机种子 13711	**总分**	0.5923	0.6335							
	AUC（源域）	0.6654	0.7140	0.9378	0.5048	0.6809	0.8493	0.7215	0.7660	0.5375
	AUC（目标域）	0.5533	0.5990	0.7977	0.5067	0.3969	0.7689	0.6300	0.5625	0.5301
	pAUC（源域与目标域）	0.5628	0.5779	0.7826	0.5035	0.5109	0.6095	0.5786	0.5440	0.5161
随机种子 13591	**总分**	0.5897	0.6303							
	AUC（源域）	0.6700	0.7155	0.9297	0.5183	0.6862	0.8504	0.7153	0.7678	0.5409
	AUC（目标域）	0.5528	0.5924	0.7547	0.5116	0.3984	0.7681	0.6254	0.5593	0.5296
	pAUC（源域与目标域）	0.5580	0.5707	0.7509	0.5047	0.5068	0.6014	0.5751	0.5395	0.5161
随机种子 13267	**总分**	0.5890	0.6262							
	AUC（源域）	0.6650	0.7138	0.9401	0.5031	0.6820	0.8510	0.7195	0.7637	0.5371
	AUC（目标域）	0.5557	0.5999	0.7965	0.5075	0.4055	0.7724	0.6280	0.5595	0.5296
	pAUC（源域与目标域）	0.5623	0.5763	0.7738	0.5067	0.5133	0.6071	0.5763	0.5408	0.5164
平均值	**总分**	**0.5903**	**0.6300**							
参见 DCASE2022T2	AUC（源域）	0.6450	0.6755	0.7753	0.5981	0.6383	0.6975	0.7271	0.7605	0.5315
基线自编码器*2	AUC（目标域）	0.4515	0.4833	0.5100	0.3919	0.3267	0.5702	0.5384	0.4959	0.5501
（不符合首次提交合规要求）	pAUC（源域与目标域）	0.5292	0.5356	0.6249	0.4944	0.4995	0.5209	0.5563	0.5391	0.5144
	总分	0.5305	0.5648							
参见 DCASE2022T2	AUC（源域）	0.5907	0.6436	0.6347	0.4692	0.7088	0.6283	0.6067	0.7470	0.7103
基线 MobileNetV2*2	AUC（目标域）	0.4748	0.5230	0.4037	0.6302	0.3506	0.4619	0.5460	0.4971	0.7715
（不符合首次提交合规要求）	pAUC（源域与目标域）	0.5358	0.5420	0.5582	0.5082	0.5191	0.4972	0.5091	0.5291	0.6726
	总分	0.5295	0.5695							
参见 DCASE2022T2 第1名	AUC（源域）	0.7734	0.8312	0.8818	0.6176	0.7435	0.9668	0.7190	0.9516	0.9380
Liu 重庆邮电大学任务2_4*2	AUC（目标域）	0.7212	0.7514	0.8851	0.7250	0.5484	0.8419	0.6827	0.7557	0.8213
[4] （大型集成，	pAUC（源域与目标域）	0.7464	0.7618	0.8845	0.7046	0.5734	0.8604	0.6885	0.7286	0.8387
不符合首次提交合规）	**总分**	**0.7097**	**0.7815**							

*1*hmean* 表示调和平均数数，而 算术均值 表示算术平均数。 *2最后三个系统不符合首样本要求 s.

五、结论

本文提出了一种适用于机器状态监测的首次提交合规无监督异常检测方案基线系统。

所提出的系统符合首样本要求；也就是说，它不会使用来自不同机器 ID 的其他数据样本，也不会使用基于对测试数据真实标签的观察结果设定的超参数或集成。该系统仅使用给定的正常训练数据进行训练。

实验结果表明，所提出的符合首次提交要求的基线自编码器和选择性马氏距离自编码器的性能优于 DCASE2022T2 基线自编码器和 MobileNetV2 等基线方法。此外，结果还表明，这些基线自编码器的性能无法与 DCASE2022T2 的第一名获胜者相提并论该获胜者采用了依赖机器类型的超参数调优和大规模工具集成。

由于所提出的系统符合首样本要求，因此预计它在工业中的任何真实数据上都能取得类似的良好表现。

所提出的系统将作为基线，为即将举行的 DCASE 2023年挑战赛任务2：面向机器状态监测的首次提交无监督异常声音检测提供性能基准。所提出系统的源代码可在 GitHub[9] 上获取。

预计在整个挑战赛期间还会进一步改进。

致谢

作者感谢 Yuri Musashijima 先生在系统实现和实验开展方面所付出的辛勤努力。

参考文献

[1]Y. Koizumi, Y. Kawaguchi, K. Imoto, T. Nakamura, Y. Nikaido, R. Tanabe, H. Purohit, K. Suefusa, T. Endo, M. Yasuda, and N. Harada, “DCASE2020 挑战赛任务2说明与讨论：机器状态监测的无监督异常声音检测”，载于《*2020*年声学场景与事件检测和分类研讨会论文集（*DCASE2020*）》，2020年11月，第81–85页。[在线]。获取地址：http://dcase.community/documents/workshop2020/proceedings/DCASE2020Workshop Koizumi 3.pdf [2] Y. Kawaguchi, K. Imoto, Y. Koizumi, N. Harada, D. Niizumi, K. Dohi, R. Tanabe, H. Purohit, and T. Endo, “dcase 2021 挑战赛任务2说明与讨论：域偏移条件下机器状态监测的无监督异常检测”，载于《第六届*2021*年声学场景与事件检测和分类研讨会论文集（*DCASE2021*）》，西班牙巴塞罗那，2021年11月，第186–190页。[3] K. Dohi, K. Imoto, N. Harada, D. Niizumi, Y. Koizumi, T. Nishida, H. Purohit, R. Tanabe, T. Endo, M. Yamamoto, and Y. Kawaguchi, “dcase 2022 挑战赛任务2说明与讨论：应用域泛化技术进行机器状态监测的无监督异常声音检测”，载于《第七届*2022*年声学场景与事件检测和分类研讨会论文集（*DCASE2022*）》，法国南锡，2022年11月。[4] Y. Zeng, H. Liu, L. Xu, Y. Zhou, and L. Gan, “机器状态监测的鲁棒异常声音检测框架”，DCASE2022 挑战赛，技术报告，2022年7月。[5] I. Kuroyanagi, T. Hayashi, K. Takeda, and T. Toda, “采用域泛化和专门化技术的两阶段异常声音检测系统”，DCASE2022 挑战赛，技术报告，2022年7月。[6] F. Xiao, Y. Liu, Y. Wei, J. Guan, Q. Zhu, T. Zheng, and J. Han, “dcase2022 挑战赛任务2系统：采用自监督属性分类和基于高斯混合模型的聚类进行异常声音检测”，DCASE2022 挑战赛，技术报告，2022年7月。[7] Y. Deng, J. Liu, and W.-Q. Zhang, “Aithu 系统：通过声音对机器运行状态进行无监督异常检测”，DCASE2022 挑战赛，技术报告，2022年7月。[8] S. Venkatesh, G. Wichern, A. Subramanian, and J. Le Roux, “用于提升无监督异常声音检测域泛化能力的解耦代理任务学习”，DCASE2022 挑战赛，技术报告，2022年7月。[9] N. Harada, Y. Musashijima, and D. Niizumi, “dcase2023 task2 基线自编码器”，2023，https://github.com/nttcslab/ dcase2023 task2 baseline ae。[10] “DCASE 2023 挑战赛任务2：机器状态监测的首次无监督异常声音检测”，2023，https://dcase. community/challenge2023/。[11] K. Dohi, T. Nishida, H. Purohit, R. Tanabe, T. Endo, M. Yamamoto, Y. Nikaido, and Y. Kawaguchi, “MIMII DG：用于域泛化任务的工业机器故障调查与检查声音数据集”，*arXiv* 预印本：*2205.13879*，2022。[12] N. Harada, D. Niizumi, D. Takeuchi, Y. Ohishi, M. Yasuda, and S. Saito, “ToyADMOS2：另一种用于域偏移条件下异常声音检测的微型机器运行声音数据集”，载于《第六届*2021*年声学场景与事件检测和分类研讨会论文集（*DCASE2021*）》，西班牙巴塞罗那，2021年11月，第1–5页。[13] K. Dohi, K. Imoto, N. Harada, D. Niizumi, Y. Koizumi, T. Nishida, H. Purohit, T. Endo, M. Yamamoto, and Y. Kawaguchi, “DCASE 2022 挑战赛任务2说明与讨论：应用域泛化技术进行机器状态监测的无监督异常声音检测”，*arXiv* 预印本：*2206.05876*，2022。[14] K. Dohi, “dcase2022 task2 基线自编码器”，2022，https://github.com/ Kota-Dohi/dcase2022 task2 baseline ae。[15] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2：倒置残差与线性瓶颈”,–*2018*年*IEEE/CVF*计算机视觉与模式识别会议，第4510–4520页，2018年。[16] K. Dohi, “dcase2022 task2 MobileNet V2 基线”，2022，https://github.com/Kota-Dohi/dcase2022 task2 baseline mobile net v2。[17] P. Mahalanobis, “统计学中的广义距离”，载于《印度国家科学院论文集》，第12期，1936年，第49–55页。[18] K. Suefusa、T. Nishida、H. Purohit、R. Tanabe、T. Endo 和 Y. Kawaguchi，“基于插值深度神经网络的异常声音检测”，载于*ICASSP 2020 - 2020*年*IEEE*声学、语音与信号处理国际会议（*ICASSP*），2020年，第271–275页。[19] Y. Koizumi、S. Saito、M. Yamaguchi、S. Murata 和 N. Harada，“用于最小化声音中基于深度神经网络的异常检测最大异常分数的批次均匀化”，载于*2019 IEEE*信号处理应用于音频与声学研讨会（*WASPAA*），2019年，第6–10页。[20] Y. Koizumi、S. Saito、H. Uematsu、Y. Kawachi 和 N. Harada，“基于深度学习和奈曼–皮尔逊引理的无监督异常声音检测”，*IEEE/ACM*音频、语音与语言处理汇刊，第27卷，第1期，第212–224页，2019年。