

# FMSG-JLESS SUBMISSION FOR DCASE 2024 TASK4 ON SOUND EVENT DETECTION WITH HETEROGENEOUS TRAINING DATASET AND POTENTIALLY MISSING LABELS

## Technical Report

Yang Xiao<sup>1</sup>, Han Yin<sup>2</sup>, Jisheng Bai<sup>2</sup>, and Rohan Kumar Das<sup>1</sup>

<sup>1</sup>Fortemedia Singapore, Singapore

<sup>2</sup>Joint Laboratory of Environmental Sound Sensing, School of Marine Science and Technology, Northwestern Polytechnical University, Xi'an, China

{xiaoyang, rohankd}@fortemedia.com, {yinhan, baijs}@mail.nwpu.edu.cn

### ABSTRACT

This report presents the systems developed and submitted by Fortemedia Singapore (FMSG) and Joint Laboratory of Environmental Sound Sensing (JLESS) for DCASE 2024 Task 4. The task focuses on recognizing event classes and their time boundaries, given that multiple events can be present and may overlap in an audio recording. The novelty this year is a dataset with two sources, making it challenging to achieve good performance without knowing the source of the audio clips during evaluation. To address this, we propose a sound event detection method using domain generalization. Our approach integrates features from bidirectional encoder representations from audio transformers and a convolutional recurrent neural network. We focus on three main strategies to improve our method. First, we apply mixstyle to the frequency dimension to adapt the mel-spectrograms from different domains. Second, we consider training loss of our model specific to each datasets for their corresponding classes. This independent learning framework helps the model extract domain-specific features effectively. Lastly, we use the sound event bounding boxes method for post-processing. Our proposed method shows superior macro-average pAUC and polyphonic SED score performance on the DCASE 2024 Challenge Task 4 validation dataset and public evaluation dataset.

**Index Terms**— sound event detection, semi-supervised learning, domain generalization, mixstyle

## 1. INTRODUCTION

Sound event detection (SED) [1–5] involves identifying sound events from acoustic signals and accurately classifying them into specific categories with timestamps, considering various acoustic environments. DCASE 2024 Task 4 [6], entitled “Sound Event Detection with Heterogeneous Training Dataset and Potentially Missing Labels,” focuses on SED. This task follows up on DCASE 2023 Task 4A [7, 8] and Task 4B [9] on the following aspects:

- DCASE 2023 Task 4A evaluated systems for detecting sound events using weakly labeled data (without timestamps) and unlabeled data in the DESED dataset [10]. The goal was to provide both event class and event time localization, despite multiple overlapping events in an audio recording.
- DCASE 2023 Task 4B evaluated systems for detecting sound events with soft labeled data from the MAESTRO dataset [11]. This task focused on exploring the significance of using soft labels for SED.

DCASE 2024 Task 4 aims to unify the setups of both the tasks of 2023 edition. Specifically, instead of training an SED model on each subtask separately with its dataset, an intriguing approach is to just train a single model on all available datasets. The goal is still to provide event classes along with their time boundaries, even with multiple overlapping events. This task explores leveraging training data with varying annotation granularity (temporal resolution, soft/hard labels). Systems will be evaluated on labels with different granularity to understand their behavior and robustness for various applications. Target classes in different datasets also differ, so sound labels present in one dataset might not be annotated in another. The systems need to handle potentially missing target labels during training and perform without knowing the origin of the audio clips at evaluation time.

Although previous years’ challenges, like the frequency dynamic convolutional recurrent neural network (FDY-CRNN) [12–14], have shown notable performance in DCASE Task 4, this year’s Task 4 introduces new challenges. The main challenge is how to combine heterogeneous training datasets from diverse domains with different annotations to improve performance. Deep neural networks struggle to generalize across diverse domains, leading to poor results in real-world scenarios. Therefore, domain generalization (DG) [15] has become an essential research topic in fields like computer vision, audio processing, and natural language processing. Inspired by the exploration in DCASE Task 1 [16], which dealt with audio clips from multiple devices, we propose using the domain generalization approach for this year’s Task 4.

In this technical report, we outline our contributions to our submission for DCASE 2024 Task 4. The primary contributions of our submissions are as follows:

- We utilize the frame-level embeddings generated by the pre-trained BEATs model in late-fusion with the FDY-CRNN and then fed into the recurrent neural network with the classifier.
- We leverage the DG to explore the appropriate way to use the heterogeneous training datasets from diverse domains.
- We modify the baseline framework to independently compute the training loss for our model, which is specific to each dataset for their corresponding classes.
- We employ the sound event bounding boxes method as a post-processing method to further enhance the performance in the DESED dataset.

## 2. DATASET

The DCASE 2024 Challenge Task 4 comprises two datasets, and participants must use both in the training phase and provide one individual model that performs well for the two datasets.

### 2.1. DESED dataset

DESED [10] consists of 10-second audio clips either recorded in a domestic environment or synthesized to reproduce such an environment. It features annotated sound events from 10 classes: alarm bell ringing, blender, cat, dishes, dog, electric shaver/toothbrush, frying, running water, speech, and vacuum cleaner. The synthetic part of the dataset is generated with Scaper [17] using foreground events from the Freesound datasets and backgrounds from YouTube videos and the Freesound subset of the MUSAN [18] dataset. The real-world recording part comes from AudioSet [19] and includes a weakly annotated set (1,578 clips), an unlabeled set (14,412 clips), and a strongly annotated portion (3,470 clips).

### 2.2. MAESTRO Real dataset

MAESTRO Real [11, 20], used in the DCASE 2023 Task 4B challenge, consists of a development set (6,426 clips) and an evaluation set of long-form real-world recordings. This dataset contains multiple temporally strong annotated events with soft labels from 17 classes. However, in this challenge, only 11 classes are evaluated, as the other 6 do not have confidence levels over 0.5. These classes are: birds\_singing, car, people\_talking, footsteps, children\_voices, wind\_blowing, brakes\_squeaking, large\_vehicle, cutlery\_and\_dishes, metro\_approaching, and metro\_leaving. This data was annotated using crowdsourcing, where temporally-weak labeling is combined with a sliding window approach to determine events' temporal localization. In order to obtain the soft labels, annotations of multiple annotators are aggregated via MACE [21]. The recordings are taken from the TUT Acoustic Scenes 2016 dataset [22] and are between 3 to 5 minutes long.

## 3. PROPOSED APPROACH

### 3.1. Baseline

The baseline system is inherited from previous DCASE Task 4 challenges [7, 23] and consists of a CRNN [24] that uses self-supervised features from the pre-trained BEATs [25] model. First, the CRNN has a convolutional neural network (CNN) encoder with 7 convolutional layers, batch normalization, gated linear units, and dropout, followed by a bi-directional gated recurrent unit (biGRU) layer. Then, BEATs features are concatenated with the CNN-extracted features before the biGRU layer. Average pooling is applied to the BEATs features to match the sequence length of the CNN encoder. Finally, Attention pooling is used to derive clip-wise and frame-wise posteriors. During training, the BEATs model remains frozen, and the mean-teacher framework [26, 27] is used to leverage unlabeled and weakly labeled data. The attention pooling mechanism in the baseline model uses the softmax function over classes. Before applying softmax, values for unlabeled classes (not in the current clip dataset) are masked to negative infinity.

As a preprocessing step, some DESED events are mapped to similar classes in MAESTRO. For example, in DESED, “speech” is a super-class for “people talking,” “children’s voices,” and “announcements” in MAESTRO. “Dishes” in DESED corresponds

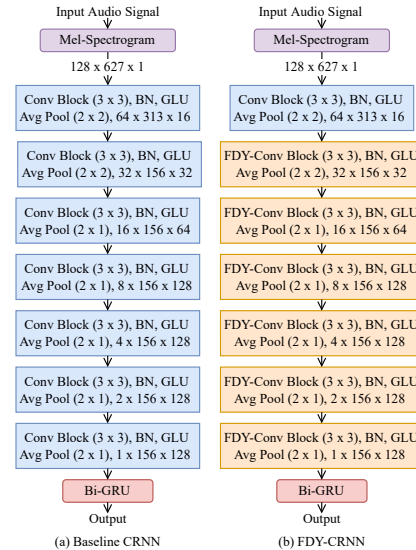


Figure 1: Architecture of (a) CRNN (Baseline) (b) FDY-CRNN. when hop length is 256.

to “cutlery and dishes” in MAESTRO, and “dog” in DESED is a super-class for “dog bark”. This mapping ensures that when computing the loss for MAESTRO, the network output for similar classes in DESED is adjusted accordingly.

### 3.2. Domain generalization

In DCASE 2024 Challenge Task 4, we must train one model on heterogeneous datasets. As mentioned in Section 2, the DESED and MAESTRO Real datasets come from different sources. Even if they share the same segment labels, their features can be quite different, confusing model training because domain information is not useful. domain generalization (DG) [15] aims to address this problem by learning robust models against data distribution changes across domains, known as domain shift. The goal is to ensure that the trained model can generalize well to any domain by learning domain-invariant feature representations that remain discriminative across multiple domains.

MixStyle [28] is a common DG method motivated by the observation that the visual domain is closely related to image style. Specifically, MixStyle mixes the feature statistics of two instances with a random convex weight to simulate new styles. However, unlike images where 2D convolution operates along spatial dimensions, in case of audio, 2D convolution operates on frequency and temporal information. Therefore, domain information may not be mainly distributed in channel statistics in audio as it is in MixStyle. In [16], the authors analyzed the relationship between the domain and the statistics of each feature dimension showing that the frequency feature dimension carries more domain-relevant information than the channel dimension. This inspired us to adopt frequency-wise MixStyle for applying it to both internal CNN features and the mel-spectrogram before feeding data into the CNN. After integrating frequency-dynamic attention and MixStyle, we propose a new model, which is introduced in the next section. We note that MixStyle is not applied during testing.

We also explored several other DG methods including the residual normalization [29], and adapted them to the audio task. However, the experiments demonstrate that the freqwise mixstyle has straightforward improvement for heterogeneous training datasets.

### 3.3. Network

In this work, we also employed FDY-CRNN from [7], which uses frequency-adaptive kernels to enforce frequency dependency in 2D convolutions. In the baseline CRNN architecture shown in Figure 1(a), we replaced the standard 2D convolutional blocks with FDY-convolutional blocks, as illustrated in Figure 1(b). The CNN part consists of 7 blocks with the same number of filters as in the baseline. In the FDY-convolutional block, batch normalization and gated linear units are used.

### 3.4. Independent loss function

The model is trained using the binary cross-entropy (BCE) loss function on DESED real-world strongly, synthetic, and weakly labeled data, as well as on MAESTRO soft-labeled data. Mean squared error (MSE) is used for the mean-teacher [26, 27] pseudo-labeling loss component, which is applied to both weak and unlabeled data from DESED. When computing the loss for both components on a particular clip, we avoid computing the loss for the network outputs corresponding to classes that do not belong to the clip’s original dataset. For example, in case of MAESTRO, we do not compute the loss for DESED output logits even for classes that have been cross-mapped as explained in the baseline. This is the main difference between our system and the baseline in loss computation.

### 3.5. Pretrained model

We utilize the pretrained BEATs model, which has achieved state-of-the-art performance on AudioSet with a mean average precision (mAP) of 0.486. The BEATs is an iterative self-supervised framework for audio representation learning, using an acoustic tokenizer and a semi-supervised learning model. Unlike previous models, BEATs employs a self-distilled tokenizer to convert audio signals into discrete labels. We use it to construct frame-level embeddings of size 768, aligning with the recently released baseline approach.

### 3.6. Data augmentation

For this year’s challenge, we used two data augmentation methods. We applied SpecAugment-style time-wise masking [30] to the features extracted by the pre-trained model and independently to the features extracted from the CNN encoder. This strategy, referred to as “dropstep”, helps improve model robustness by adding variety to the training data.

Additionally, we used the Mixup [31] strategy. This helps in linear interpolation and improves model robustness. Mixup is applied independently on the MAESTRO and DESED datasets.

### 3.7. Curated set

To solve the mismatch between the synthetic valid dataset and the real-world test set, we further split the 3,470 clips of the strongly annotated AudioSet part to get the extra real valid dataset (373 clips).

### 3.8. Sound event box-based post-processing

Existing systems [32] commonly predict sound presence confidence in short time frames. Then, thresholding produces binary frame-level presence decisions, with the extent of individual events determined by merging consecutive positive frames. In the previous challenge [7], we used median filtering as post-processing.

A recent study in [33] shows that frame-level thresholding degrades the prediction of event extent by coupling it with the system’s sound presence confidence. Inspired by bounding box predictions in image object detection [34] SEBBs are one-dimensional bounding boxes defined by event onset time, event offset time, sound class, and confidence. They represent sound event candidates with a scalar confidence score. The final SED is derived by class-wise event-level thresholding of the SEBBs’ confidences. SEBBs whose confidence exceeds the threshold are accepted as detections, while the rest are discarded. The threshold controls the sensitivity of systems. For high sensitivity/recall (few missed hits), a low detection threshold detects events even when the system’s confidence is low. For high precision (few false alarms), a higher threshold detects only events with high confidence. With SEBBs, the sensitivity of a system can be controlled without impacting the detection of an event’s onset and offset times, which was a problem with the previous frame-level thresholding approach.

We first tune the hyperparameters for the change-point-based predictor of Sound Event Bounding Boxes (cSEBBs) [33] based on the strong validation dataset, then we use cSEBBs as the post-processing method in our system.

## 4. EXPERIMENTAL SETUP

### 4.1. Feature extraction

All audio clips are resampled to a 16 kHz mono channel using Librosa. They are segmented with a window size of 2048 samples and a hop length of either 160 or 256 samples. A short-time Fourier transform is applied to extract spectrograms. Mel-filters are then used to create log-mel spectrograms spanning from 0 to 8 kHz. Clips shorter than 10 seconds are padded with silence if needed.

### 4.2. Training method

For all experiments, a batch size of 60 was used, comprising the strong set, weak set, and unlabeled set, with batch size distribution: approximately 1/5 of the maestro dataset, 1/10 of the synth dataset, 1/10 of the synth+strong dataset, 1/5 of the weak dataset, and 2/5 of the unlabeled dataset. The training process included 50 epochs for warmup, a maximum of 300 epochs, and an epoch decay of 100. Gradient clipping was set at 5.0, the EMA factor for the mean teacher [26] was 0.999, validation was performed every 10 epochs, and the maximum weight for self-supervised loss was 2. The Adam optimizer was employed with a learning rate of 0.001. An exponential warmup was applied for the initial 50 epochs, and no early stopping was implemented during the training process.

### 4.3. Evaluation metric

This year, this task requires us to consider the PSDS [35, 36] for evaluation. Event onset and offset times required for PSDS computation are only available for DESED data and classes, so PSDS is only evaluated on this fraction of the evaluation set. For MAESTRO, segment-based labels (one second) are provided, and we use the segment-based mean (macro-averaged) partial area under the ROC curve (mPAUC) as the primary metric, with a maximum FP-rate of 0.1. mPAUC is computed with respect to hard labels (threshold = 0.5) for the 11 classes listed. DESED and MAESTRO clips are anonymized and shuffled in the evaluation set to prevent manual domain identification.

Table 1: Performance in PSDS and mPAUC of different single-systems on the DESED development set (D-PSDS), DESED public evaluation set (PE-PSDS), and MAESTRO evaluation set (mPAUC) including frequency-dynamic convolution (FDY), domain generalization (DG), and strong validation dataset (Strong Val). ‘AdaResNorm’ and ‘ResNorm’ stands for adaptive residual normalization and residual normalization, respectively. Based on the joint score, the sum of PE-PSDS and mPAUC, the systems with ‘\*’ are chosen for final submission.

System	DG	Hop_length	FDY	Strong Val	D-PSDS	PE-PSDS (raw)	PE-PSDS (cSEBBs)	mPAUC	Joint score
Baseline	-	256	-	-	0.483	0.529	-	0.721	1.250
S-1*	MixStyle	256	-	-	0.506	0.587	0.629	0.737	1.366
S-2	MixStyle	256	✓	-	<b>0.510</b>	0.590	0.600	<b>0.753</b>	1.353
S-3*	MixStyle inside FDY	256	✓	-	0.503	0.590	0.634	0.737	1.371
S-4	MixStyle	256	-	✓	0.500	0.595	0.615	0.745	1.360
S-5	AdaResNorm	256	-	-	0.493	0.589	0.593	0.747	1.340
S-6	ResNorm	256	-	-	0.491	0.595	0.595	0.733	1.328
S-7*	MixStyle	160	-	-	0.480	0.588	<b>0.643</b>	0.748	<b>1.391</b>
S-8	MixStyle inside FDY	160	✓	-	0.485	<b>0.599</b>	0.629	0.737	1.366

#### 4.4. Ensemble

Ensemble modeling is a technique that leverages the strengths of multiple models to improve overall performance and enhance the generalization capability of a system. In our system, ensemble modeling plays a crucial role to improve system performance. These models work together to extract the best aspects from the highest-performing models. To generate final predictions, we aggregate the individual predictions from all the models and calculate their average. This approach ensures that every model contributes to the overall performance of the ensemble system.

### 5. RESULTS AND ANALYSIS

In this section, we first present the findings of the of the 3 single systems, followed by 1 ensemble system that we submitted.

#### 5.1. Single-systems

The PSDS measures the performance of systems on SED for the DESED subsets. In the DESED development set, the baseline system achieves a PSDS of 0.483. Systems S-1\* (0.506), S-2 (0.510), S-3\* (0.503), and S-4 (0.500) outperform the baseline, showing significance of MixStyle to improve performance. Systems S-5 (0.493) and S-6 (0.491) show slight improvements, while S-7\* (0.480) and S-8 (0.485) perform slightly worse, suggesting the hop length of 160 might be less effective. For the public evaluation set, while the baseline achieves 0.529 our systems S-1\* (0.587), S-2 (0.590), S-3\* (0.590), S-4 (0.595), and S-8 (0.599) outperform the baseline, with S-8 achieving the highest score, indicating the benefit of integrating MixStyle inside FDY.

The mPAUC is another key metric that is used for MAESTRO subset. The baseline system achieves an mPAUC of 0.721. System S-1\* shows a significant improvement with 0.737, highlighting the positive impact of MixStyle on generalization. System S-2 achieves the highest mPAUC of 0.753, which indicates combining MixStyle with frequency-dynamic convolution significantly enhances performance. System S-3\* also performs well with 0.737, reinforcing the benefits of domain generalization. Systems S-5 (0.747) and S-6 (0.733) show moderate improvements with adaptive normalization techniques, while S-7\* (0.748) demonstrates excellent performance even with a shorter hop length. System S-8, integrating MixStyle inside FDY, shows consistent improvement with an mPAUC of 0.737.

When combining results from both PSDS and mPAUC, it becomes more evident that systems integrating MixStyle and

frequency-dynamic convolution (FDY) outperform the baseline across different datasets. System S-8, for instance, achieves the highest public evaluation set PSDS (0.599) and a strong mPAUC (0.737), demonstrating its robustness and adaptability. Systems S-1\* and S-3\* also show balanced improvements across both metrics, making them reliable choices for the final submission.

We chose systems S-1\*, S-3\*, and S-7\* for final submission due to their superior performance and generalization capabilities. System S-1\* achieves high scores across all metrics, indicating robust generalization. System S-3\* combines MixStyle with frequency-dynamic convolution, effectively enhancing domain generalization and event-level detection. System S-7\* shows excellent performance with a shorter hop length, demonstrating the model’s adaptability to different configurations. These single-systems consistently outperform the baseline and showed significant improvements in terms of both the performance metrics, making them the best candidates for submission.

#### 5.2. Ensemble system

We used all the single-systems except S-5 and S-6 for the ensemble system. After applying the ensemble method mentioned in the previous section, we achieved the highest development dataset PSDS of 0.520, a raw public evaluation dataset PSDS of 0.620, and an mPAUC of 0.762. After applying cSEBBs post-processing, the public evaluation dataset PSDS improved to 0.656, and thereby improved the joint score to **1.418**. We submitted this system as our only ensemble system.

### 6. CONCLUSION

This report presents SED systems by Fortemedia Singapore and the Joint Laboratory of Environmental Sound Sensing as a participation to DCASE 2024 Task 4. We addressed the challenge of recognizing overlapping events from different sources using a method that integrates bidirectional encoder representations from audio transformers and a convolutional recurrent neural network. Our key strategies included applying MixStyle to adapt multi-domains, using an independent learning framework for dataset-specific training loss, and employing sound event bounding boxes for post-processing. As a part of our submission we submitted 3 single-systems and 1 ensemble system. Our ensemble system achieved the highest public evaluation dataset PSDS of 0.656, and mPAUC of 0.762, thereby showed significant improvement over the challenge baseline.

## 7. REFERENCES

- [1] A. Mesaros, T. Heittola, T. Virtanen, and M. D. Plumbley, "Sound event detection: A tutorial," *IEEE Signal Processing Magazine*, vol. 38, no. 5, pp. 67–83, 2021.
- [2] T. Khandelwal, R. K. Das, and E. S. Chng, "Sound event detection: A journey through DCASE challenge series," *APSIPA Transactions on Signal and Information Processing*, vol. 13, 2024.
- [3] J. P. Bello, C. Silva, O. Nov, R. L. Dubois, A. Arora, J. Salamon, C. Mydlarz, and H. Doraiswamy, "SONYC: A system for monitoring, analyzing, and mitigating urban noise pollution," *Communications of the ACM*, vol. 62, no. 2, pp. 68–77, 2019.
- [4] J. P. Bello, C. Mydlarz, and J. Salamon, "Sound analysis in smart cities," *Springer International Publishing*, pp. 373–397, 2018.
- [5] C. Debes, A. Merentitis, S. Sukhanov, M. Niessen, N. Frangiadakis, and A. Bauer, "Monitoring activities of daily living in smart homes: Understanding human behavior," *IEEE Signal Processing Magazine*, vol. 33, no. 2, pp. 81–94, 2016.
- [6] S. Cornell, J. Ebberts, C. Douwes, I. Martín-Morató, M. Harju, A. Mesaros, and R. Serizel, "DCASE 2024 task 4: Sound event detection with heterogeneous data and missing labels," *arXiv preprint:2406.08056*, 2024.
- [7] Y. Xiao, T. Khandelwal, and R. K. Das, "FMSG submission for DCASE 2023 challenge task 4 on sound event detection with weak labels and synthetic soundscapes," DCASE 2023 Challenge, Tech. Rep., 2023.
- [8] Y. Xiao and R. K. Das, "Dual knowledge distillation for efficient sound event detection," *arXiv preprint:2402.02781*, 2024.
- [9] H. Yin, J. Bai, S. Huang, and J. Chen, "How information on soft labels and hard labels mutually benefits sound event detection tasks," DCASE 2023 Challenge, Tech. Rep., 2023.
- [10] N. Turpault, R. Serizel, A. P. Shah, and J. Salamon, "Sound event detection in domestic environments with weakly labeled data and soundscape synthesis," in *Proc. the Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*, 2019.
- [11] I. Martín-Morató and A. Mesaros, "Strong labeling of sound events using crowdsourced weak labels and annotator competence estimation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 902–914, 2023.
- [12] H. Nam, S.-H. Kim, B.-Y. Ko, and Y.-H. Park, "Frequency dynamic convolution: Frequency-adaptive pattern recognition for sound event detection," in *Proc. INTERSPEECH*, 2022, pp. 2763–2767.
- [13] T. Khandelwal, R. K. Das, A. Koh, and E. S. Chng, "FMSG-NTU submission for DCASE 2022 task 4 on sound event detection in domestic environments," DCASE 2022 Challenge, Tech. Rep., 2022.
- [14] —, "Leveraging Audio-Tagging Assisted Sound Event Detection using Weakified Strong Labels and Frequency Dynamic Convolutions," in *Proc. IEEE Statistical Signal Processing Workshop (SSP)*, 2023, pp. 329–333.
- [15] K. Zhou, Z. Liu, Y. Qiao, T. Xiang, and C. C. Loy, "Domain generalization: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 4396–4415, 2022.
- [16] B. Kim, S. Yang, J. Kim, H. Park, J. Lee, and S. Chang, "Domain generalization with relaxed instance frequency-wise normalization for multi-device acoustic scene classification," in *Proc. INTERSPEECH*, 2022, pp. 2393–2397.
- [17] J. Salamon, D. MacConnell, M. Cartwright, P. Li, and J. P. Bello, "Scaper: A library for soundscape synthesis and augmentation," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2017, pp. 344–348.
- [18] D. Snyder, G. Chen, and D. Povey, "MUSAN: A music, speech, and noise corpus," *arXiv preprint:1510.08484*, 2015.
- [19] J. F. Gemmeke, D. P. 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 *Proc. IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 2017, pp. 776–780.
- [20] I. Martín-Morató, M. Harju, P. Ahokas, and A. Mesaros, "Training sound event detection with soft labels from crowdsourced annotations," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.
- [21] D. Hovy, T. Berg-Kirkpatrick, A. Vaswani, and E. Hovy, "Learning whom to trust with MACE," in *Proc. the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, 2013, pp. 1120–1130.
- [22] A. Mesaros, T. Heittola, and T. Virtanen, "TUT database for acoustic scene classification and sound event detection," in *Proc. 24th European Signal Processing Conference (EUSIPCO)*, 2016, pp. 1128–1132.
- [23] F. Ronchini, S. Cornell, R. Serizel, N. Turpault, E. Fonseca, and D. P. Ellis, "Description and analysis of novelties introduced in DCASE task 4 2022 on the baseline system," *arXiv preprint:2210.07856*, 2022.
- [24] E. Çakır, G. Parascandolo, T. Heittola, H. Huttunen, and T. Virtanen, "Convolutional recurrent neural networks for polyphonic sound event detection," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 6, pp. 1291–1303, 2017.
- [25] S. Chen, Y. Wu, C. Wang, S. Liu, D. Tompkins, Z. Chen, and F. Wei, "BEATs: Audio pre-training with acoustic tokenizers," in *Proc. International conference on machine learning (ICML)*, vol. 202, 2023, pp. 5178–5193.
- [26] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.
- [27] L. JiaKai, "Mean teacher convolution system for DCASE 2018 task 4," DCASE 2018 Challenge, Tech. Rep., 2018.
- [28] K. Zhou, Y. Yang, Y. Qiao, and T. Xiang, "Domain generalization with mixstyle," in *Proc. International Conference on Learning Representations (ICLR)*, 2020.
- [29] H. Nam and H.-E. Kim, "Batch-instance normalization for adaptively style-invariant neural networks," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 31, 2018.
- [30] 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," in *Proc. INTERSPEECH*, 2019, pp. 2613–2617.
- [31] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," in *Proc. International Conference on Learning Representations (ICLR)*, 2018.
- [32] L. Cances, P. Guyot, and T. Pellegrini, "Evaluation of post-processing algorithms for polyphonic sound event detection," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2019, pp. 318–322.
- [33] J. Ebberts, F. G. Germain, G. Wichern, and J. Le Roux, "Sound event bounding boxes," *arXiv preprint:2406.04212*, 2024.
- [34] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.
- [35] C. Bilen, G. Ferroni, F. Tuveri, J. Azcarreta, and S. Krstulovic, "A framework for the robust evaluation of sound event detection," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 61–65.
- [36] J. Ebberts, R. Haeb-Umbach, and R. Serizel, "Post-processing independent evaluation of sound event detection systems," in *Proc. the Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*, 2023.