CochlScene Pre-Training and Device-Aware Distillation for Low-Complexity Acoustic Scene Classification

Dominik Karasin[1*], *Ioan-Cristian Olariu*[1*], *Michael Schöpf* [1*], *Anna Szymańska*[1,2*]

[1]Johannes Kepler Universität Linz, Austria [2]Lodz University of Technology, Poland
{k12213736, k12219769, k12213283}@students.jku.at, 247105@edu.p.lodz.pl

***Abstract*—Acoustic Scene Classification (ASC) aims to categorize short audio clips into pre-defined scene classes. The DCASE 2025 Challenge Task 1 evaluates ASC systems on the TAU Urban Acoustic Scenes 2022 Mobile dataset, under strict low complexity constraints (128 kB memory, 30 MMACs), with only 25% of labels available and device IDs provided at inference. In this work, we present an ASC system that exploits device type information and pretraining on an external ASC dataset to improve the classification performance. In addition, we conduct an ablation study to quantify the impact of each component in our pipeline. Our approach centers on a compact CP-Mobile student model distilled via Bayesian ensemble averaging from different combinations of CP-ResNet and BEATs teachers. We evaluate domain-specific pre-training on the CochlScene dataset on both student and teachers to compensate for label scarcity. Additionally, we apply a rich data augmentation suite, of which Device Impulse Response augmentation was particularly effective. Finally, we exploit device IDs to fine-tune specialized student and teacher models per recording device. On the TAU Urban 2022 development-test dataset, our system achieved a macro-averaged accuracy of 60.5%, representing an 8.61 percentage point improvement over the DCASE baseline, securing us the top rank in the DCASE 2025 Task 1 Challenge.**

***Index Terms*—Low-complexity Acoustic Scene Classification, Knowledge Distillation, CochlScene, Bayesian Ensemble Averaging, DIR Augmentation, Freq-MixStyle, CP-ResNet, BEATs, CP-Mobile**

1. INTRODUCTION

Acoustic Scene Classification (ASC) focuses on identifying acoustic scenes from raw audio. The DCASE 2025 Task 1 [1] addresses real-world challenges, such as recording device mismatch, low-complexity constraints, and limited availability of training data, with this year's focus on device information. Using the TAU Urban Acoustic Scenes 2022 Mobile dataset (TAU22) [2], the challenge aims to classify 1-second audio clips into 10 predefined audio scenes. The contestants face two low-complexity constraints: maximum memory allowance for model parameters equal to 128 kB and computational complexity at inference time restricted to 30 MMACs.

This year's focus is put on device information, which can be used to fine-tune the models for specific recording devices. Due to availability of recording device information in the evaluation dataset, distinct models can be used per device, while still applying the general model to unseen recording devices. An additional change was made in terms of the availability of data. Training data is restricted to 25% subset of the DCASE24 Task 1 dataset. However, it is permitted to utilize external ASC datasets for model development.

This paper contributes to the research on the practical application of ASC systems by studying the effect of pre-training student and teacher models on CochlScene, applying Device Impulse Response augmentation (DIR) and fine-tuning them on the recording devices. The proposed system achieved the first rank in Task 1 of the DCASE 2025 Challenge [1].

We review related work in Section 2, followed by the methodology in Section 3. Section 4 is devoted to the experimental setup, while Section 5 presents the results and discussion. Finally, the paper is concluded in Section 6.

[*]These authors contributed equally to this work.

2. RELATED WORK

We present advancements and techniques in the field of ASC, that this work build upon.

2.1. Architectures

Self attention architectures have rapidly advanced ASC by capturing long-range dependencies in time-frequency representations [3]. The Audio Spectrogram Transformer (AST) [4] introduces a fully-attention based encoder that outperforms CNNs on Audioset and ESC-50 [5] benchmarks. Building on AST, PaSST [3] employs Patchout to accelerate training and reduce redundancy, achieving SOTA results on AudioSet. More recently, BEATs [6] leaverages self-supervised pre-training with an acoustic tokenizer to learn robust representations, reaching 50.6% mAP on AudioSet-2M [7] and 98.1% accuracy on ESC-50. In DCASE Challenge submissions, these large transformer are commonly used as teachers for knowledge distillation into lightweight student models [8].

2.2. Data Augmentation

To mitigate overfitting and enhance robustness in low-data scenarios, ASC systems commonly employ various data augmentation techniques that aims to improve overall generalization [9]. At the waveform level, temporal shifts, also called time-rolling, randomly circularly shift the audio to learn temporal invariance in scene cues. Device Impulse Response (DIR) augmentation [10] convolves the input with the microphone impulse responses, to simulate audio recorded by different microphones [10]. In the time-frequency domain, SpecAugment [11] applies random masks along both time and frequency axes of the log-mel spectrogram. This prevents reliance on narrow spectro-temporal features and therefore increases robustness.

2.3. Device-aware fine-tuning

To explicitly account for device-specific distortions, models can be finetuned using device metadata or specialized modules. A simple approach is per-device fine-tuning, where a shared backbone is adapted separately on each device's data, yielding better performance on that device at inference time [12]. More parameter-efficient methods insert small adapter layers or conditional normalization into a network (e.g., FiLM [13] or device-conditional BatchNorm [14]), where only these modules are trained per device. Domain generalization techniques such as MixStyle [15] or Freq-MixStyle [16] blend feature statistics across device domains during training, which results in improved generalization and robustness for unseen devices.

3. METHODOLOGY

In this section, we describe the different components of the training pipeline used for the experiments outlined in Section 4.

3.1. Datasets

3.1.1. TAU Urban Acoustic Scenes 2022 Mobile dataset: Our primary dataset is the TAU Urban Acoustic Scenes 2022 Mobile dataset (TAU22) [2], an extension of the 2020 Mobile dataset [17]. In TAU22, each original 10-second clip has been split into ten 1-second, single-channel samples at 44.1 kHz. TAU22 includes recordings from multiple European cities across ten scene classes, captured with four real devices (A, B, C, and D) and supplemented by simulated devices (S1–S10). The ten classes in TAU22 are: *airport, bus, metro, metro station, park, public square, shopping mall, street pedestrian, street traffic, tram*.

The 2025 Low-Complexity Acoustic Scene Classification task provides both official development and evaluation splits. For development, only 25% of the official training set is permitted during model training [1]. This corresponds to last year's 25% train split. The development set can be further split into:

- **Development-train:** devices A, B, C and simulations S1–S3 (8.25 hours of audio)
- **Development-test:** devices A, B, C and simulations S1–S6 (9.7 hours of audio)

3.1.2. CochlScene dataset: CochlScene [18] is an acoustic scene dataset, collected through crowdsourcing. It consists of 76,115 single-channel audio files with a sample rate of 44.1kHz and a length of 10 seconds. There are a total of 13 different classes, spanning acoustic scenes from urban areas in South Korea. The 13 classes in CochlScene are: *bus, cafe, car, crowded indoor, elevator, kitchen, park, residential area, restaurant, restroom, street, subway, subway station*.

3.1.3. AudioSet: AudioSet [7] is a large-scale multi-label audio event dataset, gathered from YouTube. It contains over 2 million ten-second audio clips, annotated by humans across a total of 632 classes. Each sample is a single-channel audio file with a sample rate of 44.1kHz. The labels are hierarchically structured, such that categories can be subdivided into increasingly specific event labels. It is widely used as a benchmark for multi-label audio tagging, sound event detection, and pre-training of general-purpose audio feature extractors [3], [4], [19], [20].

3.2. Architectures

3.2.1. Teacher models: As previously demonstrated, CP-ResNet architecture performs effectively on the TAU22 development set [8]. In our work, the following teacher architectures were employed:

- **CP-ResNet** [21], a receptive field regularized convolutional neural network (RFR-CNN) [22], whose controlled receptive field leads to enhanced generalization for ASC.
- **BEATs** [6], an iterative audio pre-training framework to learn Bidirectional Encoder representation with Audio Transformers.

3.2.2. Student model: We employed the compact CP-Mobile (CPM) architecture [12] as the student model. The detailed architecture can be seen in the Table 1.

The architecture of CP-Mobile consists of three different blocks composed of sequences of three layers: point-wise expansion, depth-wise convolution, and point-wise projection. Each layer consists of a convolutional operation with batch normalization [23] and ReLU [24] activation applied. This structure allows to keep the expressiveness, while reducing the computational complexity of the model.
The mentioned blocks can be described as:

1) **Transition block (CPM block T)**, which increases the channel dimension and does not contain any residual connections.
2) **Standard block (CPM block S)**, which does not change the channel dimension and uses the residual connection.

Blocks	Input shape	Parameters	MACs
Initial convolutions	[1, 1, 256, 65]	2,456	2,810,960
Block 1 (CPM-S)	[1, 32, 64, 17]	4,992	5,083,456
Block 2 (CPM-D)	[1, 32, 64, 17]	4,992	5,083,456
Block 3 (CPM-S)	[1, 32, 64, 17]	4,992	3,739,968
Block 4 (CPM-T)	[1, 32, 64, 9]	6,576	2,378,096
Block 5 (CPM-S)	[1, 56, 32, 9]	15,112	4,182,352
Block 6 (CPM-T)	[1, 56, 32, 9]	20,968	5,841,328
Final convolution	[1, 104, 32, 9]	1,060	299,540

Table 1: CP-Mobile architecture indicating input shape, total parameters, and MACs per block. (CPM-S: Standard block, CPM-T: Transition block, CPM-D: Spatial Downsampling block)

3) **Spatial Downsampling block (CPM block D)**, which does not change the channel dimension and uses the residual connection with average pooling.

The structure of each block can be seen in Figure 1.

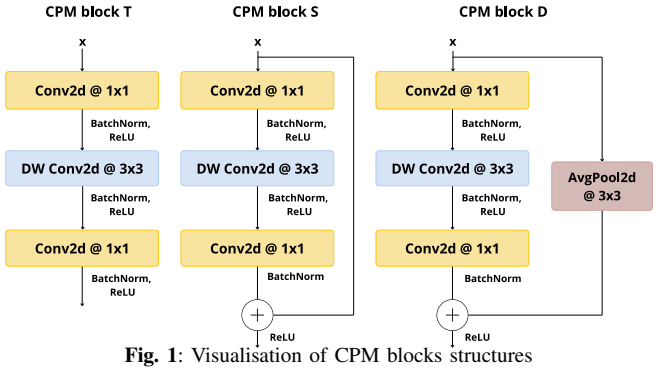

Fig. 1: Visualisation of CPM blocks structures

The first two layers of CP-Mobile project the input data from mel spectrograms to models feature space. At the last three layers, 1x1 convolution, batch normalization, and adaptive average pooling are applied.

With 61,148 parameters and 29,419,156 MACs the architecture meets the constraints when the model weights are converted to half-precision (16 bit) floating point representation for inference.

3.3. Feature extraction and data augmentation

3.3.1. Preprocessing: We resample audio to a model-specific target sampling rate and compute log-scaled mel spectrograms. The parameters used for the STFT and log-scaled mel spectrogram vary between architectures and are listed in Table 2.

Parameter	CP-Mobile	CP-ResNet	BEATs
Original sample rate (kHz)	44.1	44.1	44.1
Target sample rate (kHz)	32	32	16
FFT size	4096	4096	1024
Window length (ms)	96	96	25
Hop length (ms)	16	24	10
Number of mel bins	256	256	128

Table 2: Preprocessing parameters for different model architectures

3.3.2. Data augmentations: Both waveform-level (time rolling, Device Impulse Response (DIR) [10]) and spectrogram-level (SpecAugment [11], Freq-MixStyle [16]) augmentations were employed.

Among these, the DIR augmentation proved to strongly improve the overall accuracy of the model during the challenge. We perform DIR augmentation by convolving the audio waveform with one of 66

impulse responses taken from MicIRP[1]. The augmentation is applied with a probability of 70% to samples recorded with device A.

3.4. Pre-Training on AudioSet and CochlScene

Considering the limited size of the development dataset, previous work [25] has shown that it is beneficial to pre-train (or use existing pre-trained weights for) both the teachers and the student models on external audio datasets.

For the transformer based BEATs architecture, we use a publicly available checkpoint pre-trained on AudioSet. Since the classes and their number do not match the downstream training, the classification head is discarded. We use the checkpoint[2] provided by the authors of [20], corresponding to a model pre-trained using self-supervised learning with patch-wise masked prediction on AudioSet and fine-tuned on AudioSet with weak labels.

We furthermore use the CochlScene dataset [18] to pre-train the models involved in the preparation of the submitted systems. Since this dataset was specifically created for ASC tasks, albeit under very different urban conditions (Asia - South Korea) and using more heterogeneous recording devices, we hypothesize that models pre-trained on it would more effectively adapt to the task at hand and generalize better to unseen recording devices. The CP-ResNet teacher and the CP-Mobile student models are trained on 1-second slices of CochlScene audio clips. For BEATs teacher, we use the full audio clips, matching the 10s input size of their AudioSet pre-training. Table 3 details the key hyperparameters used.

3.5. Knowledge distillation

Knowledge distillation (KD) [26] is a training method, where a model is not only trained on the one-hot encoded class labels directly, but also on the logits of one ore more teacher models. The teachers are usually large models with high performance. Knowledge distillation in general leads to better-performing and more robust models.

Through a division of the outputs of the teacher and student models with a temperature value (τ) and subsequent application of the softmax function, softer, more informative targets are produced.

The loss function is a weighted average of a label loss (L_l), in our case the cross-entropy-loss, and the distillation loss (L_{KD}), which is the Kullback-Leibler (KL) divergence between teacher and student logits.

With λ as the weight and z_S and z_T as the output logits of the student and teacher model, the loss function is calculated as follows:

$$\text{Loss} = \lambda L_l(\delta(z_S), y) + (1-\lambda)\tau^2 L_{kd}(\delta(z_S/\tau), \delta(z_T/\tau))$$

Instead of a single teacher, we use Bayesian Ensemble Averaging (BAE) [25], [27] of several teacher models. With this, multiple teachers, possibly trained with different configurations, can be combined.

We use online KD to also apply the same data augmentation pipeline for the teacher models [28].

3.6. Device-specific training

DCASE'25 Task 1 focuses on fine-tuning the obtained model per device present during training (development-train devices: A, B, C, S1, S2, S3). The general model, for both student and teacher models, is used to initialize six specialized models, which are further fine-tuned on data specific to only one device. At inference time, the input is dispatched to a specialized model using the device ID—if known—otherwise to the general model. This way, one can obtain higher accuracies for devices encountered during training.

[1]https://micirp.blogspot.com

[2]https://github.com/fschmid56/PretrainedSED/releases

4. EXPERIMENTS

In order to evaluate the effectiveness of our proposed method, we perform two complementary sets of experiments. First, we perform step-by-step ablation study to quantify the impact and interference of three components: pre-training on CochlScene, DIR augmentation, and device-specific fine-tuning procedure. Second, we investigate different teacher model combinations and the corresponding CP-Mobile student model performance.

4.1. Training

For all experiments, we use the AdamW [29] optimizer and a cosine learning rate scheduler with the corresponding hyperparameters adapted to each task. The specific values are captured in Table 3.

Task	Dataset	Max LR	Warm-up	Epochs	Batch size
CPM pre-train	CochlScene (1s)	0.005	2000	100	512
CPM general	TAU22	0.005	2000	150	256
CPM device specific	TAU22	0.0005	200	50	256
CP-ResNet pre-train	CochlScene (1s)	0.001	2000	150	512
CP-ResNet general	TAU22	0.001	2000	100	256
CP-ResNet device-specific	TAU22	0.00001	200	50	256
BEATs pre-train	CochlScene (10s)	0.00007	5000	30	10
BEATs general	TAU22	0.00001	2000	30	80
BEATs device-specific	TAU22	0.000005	2000	30	256

Table 3: Hyperparameters used for different training tasks.

Despite the different input lengths for the BEATs model, we opt to directly use the 1-second TAU22 samples without additional adaptation. This approach results in only a minor decrease in classification accuracy while significantly reducing both training time and memory consumption.

4.2. Ablation study

We evaluate three main enhancements applied to the general model: DIR augmentation, pretraining on the CochlScene dataset, and the combination of both. These experiments are performed across three models: CP-ResNet, BEATS, CP-Mobile. A general CP-ResNet model and CP-Mobile are trained from scratch, while for the BEATs we use a checkpoint pre-trained on AudioSet. For each, the general model is trained on the 25% labeled subset of TAU22. To indicate the performance clearly, we do not use Knowledge distillation in this training procedure and the hyperparameters are kept constant. We store the variant with the highest accuracy on the TAU22 development-test dataset for each model and treat it as a starting point in the fine-tuning step. This training procedure clearly indicates the individual impact of the mentioned enhancements.

4.3. Evaluation of Teacher Model Combinations

For this set of experiments, different combinations of teachers are investigated. We perform Knowledge distillation from the following teachers: CP-Resnet, BEATs and device-specific CP-ResNet into the CP-Mobile student model. The device-specific BEATs model was not considered, since it resulted in a large increase in training time and computational cost. We then evaluated five combinations: each of the three teachers by themselves, only the general teachers and an ensemble of all three. All hyperparameters are kept constant for all combinations of the teachers. For training the general model, we use the temperature $\tau = 2$ and the weight $\lambda = 0.02$—values that produced good results in previous editions of the task [12]. For device-specific training, the best results were achieved with $\lambda = 0.1$. This procedure is meant to quantify the effect of device-specific teachers on the performance of the student model.

5. RESULTS & DISCUSSION

In this section, we provide the results of the previously described experiments. All reported results represent the mean performance aggregated across three independent runs.

5.1. Ablation study

Table 4 and Figure 2 present the results of our ablation study, in which we isolate the contribution of key components: Device Impulse Response augmentation, pretraining on the CochlScene dataset, and device-specific fine-tuning.

For the CP-Mobile student model, both DIR augmentation and CochlScene pretraining individually improved performance over the base model, yielding gains of $3.4\%pt.$ and $3.7\%pt.$ respectively. When combined, these enhancements resulted in an improvement of $5.9\%pt.$

CP-ResNet showed a similar trend: pre-training on CochlScene led to a clear gain over the base model of $2.8\%pt.$, while DIR augmentation alone slightly decreased performance by $1\%pt.$ The combination of DIR and CochlScene, however, improved the performance by $3.6\%pt.$ for this model.

Interestingly, The BEATs teacher model achieved the highest base performance, but exhibited a decrease of $1.2\%pt.$ when pre-trained on CochlScene. Applying DIR augmentation led to a slight improvement of $0.5\%pt.$ For this architecture, the combination of these two components performed $1.0\%pt.$ worse compared to the base model.

Applying device-specific fine-tuning always resulted in an improvement in performance. Specifically, $1.0\%pt.$ for CP-Mobile, $0.8\%pt.$ for CP-ResNet, and $0.7\%pt.$ for BEATs.

These results demonstrate that augmentation and pre-training strategies must be carefully tailored to the model architecture. When chosen appropriately, these methods can lead to substantial performance gains.

Method	CP-Mobile	CP-ResNet	BEATs
Base	0.513 ± 0.004	0.552 ± 0.004	0.582 ± 0.003
DIR	0.547 ± 0.003	0.542 ± 0.007	$\mathbf{0.585 \pm 0.005}$
Cochl	0.550 ± 0.003	0.580 ± 0.006	0.568 ± 0.001
DIR+Cochl	$\mathbf{0.572 \pm 0.002}$	$\mathbf{0.588 \pm 0.002}$	0.570 ± 0.001
Best+Device-Specific	0.582 ± 0.003	0.596 ± 0.006	0.592 ± 0.006

Table 4: Macro-average accuracy of the ablation study described in section 4.2 (mean±std). The best-performing model for each architecture—which is used as the basis for the device-specific model—is highlighted.

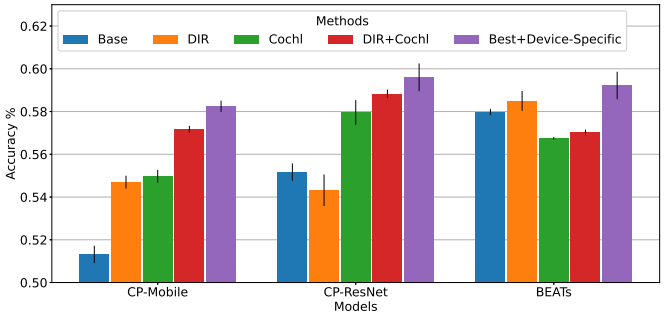

Fig. 2: Macro-average accuracy of the ablation study described in Section 4.2. The error bars show the standard deviation.

5.2. Evaluation of Teacher Model Combinations

Table 5 and Figure 3 summarize the performance of different teacher model configurations used for KD, evaluated with both general and device-specific CP-Mobile student models.

Using a single device-specific CP-ResNet teacher improves the performance of the general student by $0.4\%pt.$ compared to using a general teacher. The device-specific student improves by $0.3\%pt.$

BEATs on its own performs worse than the two CP-ResNet teachers, but using it in an ensemble with CP-ResNet improved the performance by $1.9\%pt.$ for the general and $1.4\%pt.$ for the device-specific student, compared to just using CP-ResNet.

Adding the device-specific CP-Resnet to the previously mentioned ensemble increased the macro-averaged accuracy of the general model by $0.5\%pt.$ The device-specific model improved by $0.7\%pt.$

This clearly shows how incorporating device-specific teachers can improve the performance of the student model.

Model	General	Device-Specific
BEATs	0.545 ± 0.005	0.554 ± 0.018
CP-ResNet	0.574 ± 0.008	0.582 ± 0.015
CP-ResNet DS	0.578 ± 0.009	0.585 ± 0.015
CP-ResNet + BEATs	0.593 ± 0.003	0.596 ± 0.002
BEATs + CP-ResNet + CP-ResNet DS	0.598 ± 0.003	0.603 ± 0.007

Table 5: Macro average accuracy of the CP-Mobile student trained with the given teacher combination, as described in Section 4.3 (mean±std). DS symbolizes a device-specific teacher.

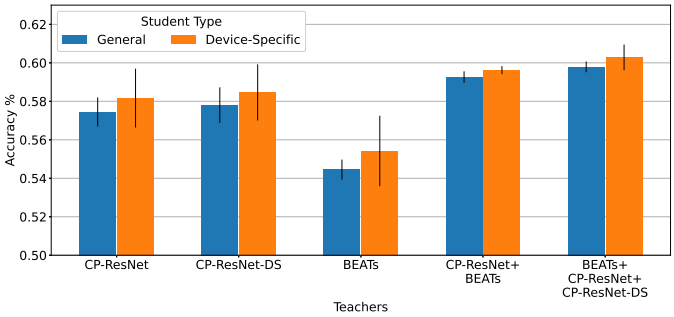

Fig. 3: Macro-average accuracy of the CP-Mobile student trained with the given teacher combination, as described in Section 4.3. DS symbolizes a device-specific teacher. The error bars show the standard deviation.

6. CONCLUSION

In this paper we presented a comprehensive approach for low-complexity Acoustic Scene Classification with the constraints for DCASE 2025 Challenge Task 1 Furthermore, we conducted an ablation study that reveals the significance of DIR, pretraining on the CochlScene dataset, and per-device fine-tuning. We conclude that the efficiency of applying DIR augmentation and CochlScene pre-training varied by model architecture. We also showed how device-specific teachers can improve the students performance during knowledge distillation. Using these methods we achieved a macro-averaged accuracy of 60.3% on the systems on the TAU Urban Acoustic Scenes 2022 Mobile dataset validation set—an 8.41 percentage point improvement over the baseline.

In the challenge setting, the best-performing model from multiple runs was selected. This accounts for the slight difference of 0.2% between the 60.5% achieved in the challenge[3] and the results reported in the experiments above.

7. ACKNOWLEDGMENT

We would like to express our gratitude to Paul Primus and Florian Schmid for their continuous support and guidance. Their feedback and expertise greatly contributed to the quality of the work. The computational results have been achieved using the Austrian Scientific Computing (ASC) infrastructure. Thanks to JKU Institute of Computational Perception for computing resources.

[3]https://dcase.community/challenge2025/task-low-complexity-acoustic-scene-classification-with-device-information-results

REFERENCES

[1] F. Schmid, P. Primus, T. Heittola, A. Mesaros, I. Martín-Morató, and G. Widmer, "Low-complexity acoustic scene classification with device information in the dcase 2025 challenge," 2025.

[2] I. Martín-Morató, F. Paissan, A. Ancilotto, T. Heittola, A. Mesaros, E. Farella, A. Brutti, and T. Virtanen, "Low-complexity acoustic scene classification in dcase 2022 challenge," in *Proceedings of the 7th Detection and Classification of Acoustic Scenes and Events 2022 Workshop (DCASE2022)*, November 2022.

[3] K. Koutini, J. Schlüter, H. Eghbal-zadeh, and G. Widmer, "Efficient Training of Audio Transformers with Patchout," in *Interspeech 2022*, Sep. 2022, pp. 2753–2757.

[4] Y. Gong, Y. Chung, and J. R. Glass, "AST: audio spectrogram transformer," in *22nd Annual Conference of the International Speech Communication Association, Interspeech 2021, Brno, Czechia, August 30 - September 3, 2021*. ISCA, 2021, pp. 571–575.

[5] K. J. Piczak, "Esc: Dataset for environmental sound classification," in *Proceedings of the 23rd ACM International Conference on Multimedia*, ser. MM '15. New York, NY, USA: Association for Computing Machinery, 2015, p. 1015–1018. [Online]. Available: https://doi.org/10.1145/2733373.2806390

[6] S. Chen, Y. Wu, C. Wang, S. Liu, D. Tompkins, Z. Chen, W. Che, X. Yu, and F. Wei, "BEATs: Audio Pre-Training with Acoustic Tokenizers," in *Proceedings of the 40th International Conference on Machine Learning*. PMLR, Jul. 2023, pp. 5178–5193.

[7] J. F. Gemmeke, D. P. W. 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 *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Mar. 2017, pp. 776–780.

[8] F. Schmid, T. Morocutti, S. Masoudian, K. Koutini, and G. Widmer, "Distilling the knowledge of transformers and CNNs with CP-mobile," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2023 Workshop (DCASE2023)*, 2023, pp. 161–165.

[9] F. Schmid, P. Primus, T. Heittola, A. Mesaros, I. Martín-Morató, K. Koutini, and G. Widmer, "Data-efficient low-complexity acoustic scene classification in the dcase 2024 challenge," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2024 Workshop (DCASE2024)*, Tokyo, Japan, October 2024, pp. 136–140.

[10] T. Morocutti, F. Schmid, K. Koutini, and G. Widmer, "Device-robust acoustic scene classification via impulse response augmentation," in *31st European Signal Processing Conference (EUSIPCO 2024)*, 09 2023, pp. 176–180.

[11] 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 *Interspeech 2019*, 2019, pp. 2613–2617.

[12] F. Schmid, T. Morocutti, S. Masoudian, K. Koutini, and G. Widmer, "Distilling the knowledge of transformers and CNNs with CP-mobile," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2023 Workshop (DCASE2023)*, 2023, pp. 161–165.

[13] E. Perez, F. Strub, H. de Vries, V. Dumoulin, and A. Courville, "FiLM: visual reasoning with a general conditioning layer," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, ser. AAAI'18/IAAI'18/EAAI'18, Feb. 2018, pp. 3942–3951.

[14] H. de Vries, F. Strub, J. Mary, H. Larochelle, O. Pietquin, and A. Courville, "Modulating early visual processing by language," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17, Red Hook, NY, USA, Dec. 2017, pp. 6597–6607.

[15] K. Zhou, Y. Yang, Y. Qiao, and T. Xiang, "Domain Generalization with MixStyle," *CoRR*, vol. abs/2104.02008, Apr. 2021.

[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 *Interspeech 2022*, 2022, pp. 2393–2397.

[17] T. Heittola, A. Mesaros, and T. Virtanen, "Acoustic scene classification in dcase 2020 challenge: generalization across devices and low complexity solutions," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2020 Workshop (DCASE2020)*, 2020, pp. 56–60.

[18] I.-Y. Jeong and J. Park, "CochlScene: Acquisition of acoustic scene data using crowdsourcing," *CoRR*, vol. abs/2211.02289, Nov. 2022.

[19] F. Schmid, K. Koutini, and G. Widmer, "Dynamic Convolutional Neural Networks as Efficient Pre-Trained Audio Models," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 32, pp. 2227–2241, Mar. 2024.

[20] F. Schmid, T. Morocutti, F. Foscarin, J. Schlüter, P. Primus, and G. Widmer, "Effective pre-training of audio transformers for sound event detection," in *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2025, pp. 1–5.

[21] K. Koutini, H. Eghbal-zadeh, and G. Widmer, "Receptive Field Regularization Techniques for Audio Classification and Tagging With Deep Convolutional Neural Networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 1987–2000, 2021.

[22] K. Koutini, H. Eghbal-zadeh, M. Dorfer, and G. Widmer, "The Receptive Field as a Regularizer in Deep Convolutional Neural Networks for Acoustic Scene Classification," in *2019 27th European Signal Processing Conference (EUSIPCO)*, Sep. 2019, pp. 1–5.

[23] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ser. ICML'15. JMLR.org, 2015, p. 448–456.

[24] A. F. Agarap, "Deep learning using rectified linear units (relu)," 2019. [Online]. Available: https://arxiv.org/abs/1803.08375

[25] D. Nadrchal, A. Rostamza, and P. Schilcher, "Data-efficient acoustic scene classification with pre-training, bayesian ensemble averaging, and extensive augmentations," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2024 Workshop (DCASE2024)*, October 2024, pp. 91–95.

[26] G. Hinton, O. Vinyals, and J. Dean, "Distilling the Knowledge in a Neural Network," in *NIPS Deep Learning and Representation Learning Workshop*, 2015.

[27] J. Xu, S. Li, A. Deng, M. Xiong, J. Wu, J. Wu, S. Ding, and B. Hooi, "Probabilistic Knowledge Distillation of Face Ensembles," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2023, pp. 3489–3498.

[28] L. Beyer, X. Zhai, A. Royer, L. Markeeva, R. Anil, and A. Kolesnikov, "Knowledge distillation: A good teacher is patient and consistent," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*. IEEE, 2022, pp. 10 915–10 924.

[29] I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization," in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*, May 2019.