HIERARCHICAL ACOUSTIC SCENE CLASSIFICATION WITH KNOWLEDGE DISTILLATION AND PRE-TRAINED DYNAMIC NETWORKS

Technical Report

Jisheng Bai[1,2,3]*, Mou Wang*[4]*, Ee-Leng Tan*[3]*, Jin Jie Sean Yeo*[3]*,*
Yeow Jun Wei[3], Santi Peksi[3], Dongyuan Shi[3], Woon-Seng Gan[3], Jianfeng Chen[1]

[1] Joint Laboratory of Environmental Sound Sensing, School of Marine Science and Technology, Northwestern Polytechnical University, Xi'an, China
[2] LianFeng Acoustic Technologies Co., Ltd., Xi'an, China
[3] Smart Nation TRANS Lab, Nanyang Technological University, Singapore
[4] Institute of Acoustics, Chinese Academy of Sciences, Beijing, China
baijs@mail.nwpu.edu.cn

ABSTRACT

Previous acoustic scene classification (ASC) tasks in the DCASE challenge focused on two important aspects: recording device mismatch and low-complexity systems. However, implementing ASC systems in real-life applications remains challenging due to the time-consuming process of collecting large amounts of labeled data for system development. DCASE2024 Task 1 is proposed to explore possible solutions that can efficiently utilize varying ratios of available training data while maintaining ASC performance. In this paper, we propose a hierarchical learning-based method to develop ASC systems using knowledge distillation and pre-trained dynamic networks. Specifically, we fine-tune the dynamic networks, which are pre-trained on Audioset, with an additional classification task and various data augmentation methods. We then employ an ensemble of fine-tuned dynamic networks to teach CP-Mobile networks. Finally, we fine-tune the CP-Mobile networks using quantization-aware training to achieve low-complexity models. The experimental results demonstrate that the proposed systems outperform the baseline system.

Index Terms— Hierarchical learning, data augmentation, dynamic CNN, knowledge distillation

1. INTRODUCTION

Acoustic scene classification (ASC) is a crucial research problem in computational auditory scene analysis that aims to recognize the unique acoustic characteristics of an environment [1]. Previous DCASE ASC tasks have focused on the challenges of domain shift in recording devices and the development of low-complexity ASC systems [2]. Although substantial progress has been made in related fields, challenges still hinder the application of ASC systems in the real world. One significant challenge is the limited availability of labeled data when developing ASC systems, as it is expensive to collect many acoustic scene recordings. Therefore, it is essential to study possible ways to use labeled data efficiently.

Deep learning algorithms have emerged as the predominant approach, significantly enhancing ASC performance [3, 4]. Deep learning-based ASC methods typically require substantial data to achieve leading performance. Considering the scarcity of labeled data, Bai et al. proposed a challenge to study potential semi-supervised learning methods for leveraging both labeled and unlabeled data in ASC [5]. DCASE2024 Task 1 has also been proposed to study the data-efficient problem of ASC by limiting different ratios of available training data [6]. The developed ASC system must be robust to unseen recording devices that are not available in the training splits. Moreover, the ASC system must be of low complexity to facilitate deployment on embedded devices. The limitations of available training data and model complexity make it challenging to develop a robust ASC system.

Pre-trained audio neural networks have emerged as a powerful solution to the challenge of insufficient labeled training data in several audio tasks [7, 8, 9]. These pre-trained models leverage vast amounts of weakly labeled audio data during their pre-training phase, enabling them to learn robust and generalized audio representations. Consequently, when fine-tuned on other smaller datasets, they demonstrate significantly improved performance and efficiency compared to models trained from scratch. However, these pre-trained audio models are usually large-scale, and even the state-of-the-art lightweight pre-trained models can not meet the complexity requirement of edge devices.

Knowledge distillation (KD) offers a promising approach to address the issue of implementing large-scale pre-trained audio models for ASC. By transferring knowledge from a large teacher model to a smaller, more efficient student model, KD enables the deployment of high-performance ASC systems on the edges. Schmid et al. trained CP-Mobiles using offline KD from an ensemble of 6 different Patchout FaSt Spectrogram Transformer (PASST) models for ASC, achieving the top performance on the DCASE2023 ASC Task [10].

In addition, some techniques have also been proven to be effective in ASC systems, including data augmentation, model compression, device generalization, and hierarchical learning (HL) [11]. Particularly, previous works introduce the hierarchical taxonomy for ASC, where the acoustic scene classes can be coarsely grouped into indoor, outdoor, and transportation[11, 12]. The model can learn experience from the high-level coarse-grained acoustic classes.

This technical report introduces an HL-based ASC system with KD and pre-train dynamic CNNs. We first fine-tune the dynamic

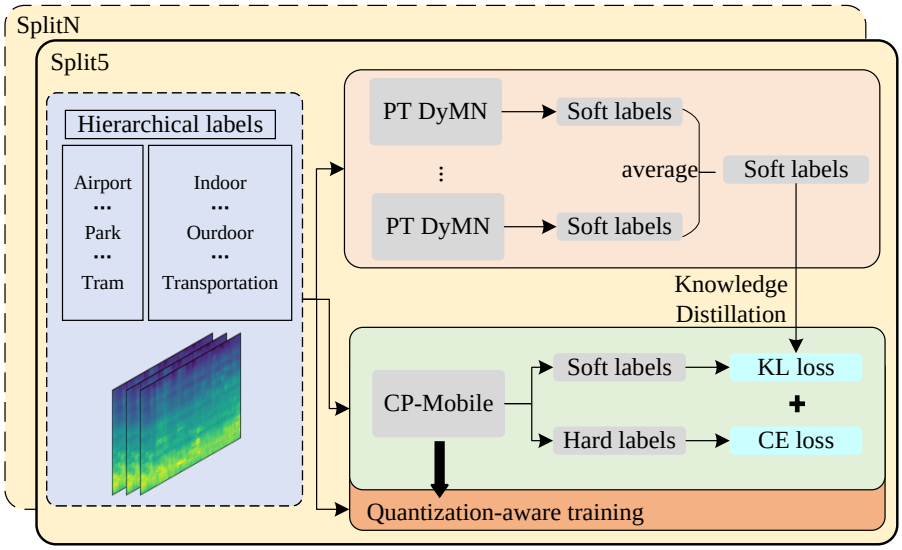

Figure 1: The overall framework of the proposed method. PT DyMN refers to pre-trained dynamic MobibleNet.

CNNs pre-trained on Audioset using each split of the development dataset. We apply data augmentation, device generalization methods, and 2 different HL methods during training to further improve performance. The HL methods aim to additionally classify the 10 acoustic scene classes into 3 different high-level semantic classes: indoor, outdoor, and transportation. Next, we train CP-Mobile networks using offline KD by assembling the teacher models. Finally, we use the same KD procedures to fine-tune the CP-Mobile networks with quantization-aware training (QAT) to further compress the student model. Experimental results show that the proposed method outperforms the baseline and fits within the complexity limitations.

The remainder of this paper is structured as follows: Section 2 introduces the details of the proposed framework. Section 3 describes the experimental settings results. Section 4 concludes this technical report.

2. PROPOSED METHOD

In this section, we first introduce the overall framework of the proposed method. Then we introduce the model architectures, HL, QAT, data augmentation, and device generalization methods used in our systems.

2.1. Overall framework

The whole framework is shown in Fig 1. For each split of the development dataset, we repeat the same procedures:

1 We fine-tune the pre-trained dynamic neural networks by incorporating 2 HL methods. We also apply data augmentation and device generalization methods during the fine-tuning. We average the logits of the fine-tuned dynamic neural networks and use them to teach student models.

2 We train the CP-Mobile networks from scratch using the averaged logits from teacher models. We also incorporate the HL method, data augmentation, and device generalization methods during this stage.

3 We fine-tune the CP-Mobile networks using QAT. We also keep distilling the knowledge into the CP-Mobile networks and incorporating multi-task learning, data augmentation, and device generalization methods. Finally, we save the quantized CP-Mobile networks for evaluation.

2.2. Model architecture

2.2.1. Dynamic CNNs

Pre-trained neural networks can significantly enhance the performance on downstream tasks. These networks are more efficient when the data of downstream tasks is insufficient for training the model from scratch. CNN-based pre-trained audio neural networks were first proposed and achieved promising performance on many audio tasks[7]. Then Transformer-based pre-trained audio neural networks outperform CNN ones while introducing more computation cost [8]. Recently, Schmid et al. introduced dynamic components into the MobileNets and used KD from a series of pre-trained large-scale Transformer-based model assemblies for training [9]. These dynamic CNNs are more efficient and achieve competitive performance with Transformer-based pre-trained audio neural networks. To this end, we introduce the pre-trained dynamic CNNs as our efficient teacher models for ASC.

Schmid et al. designed the architectures of dynamic CNNs following the MobileNetV3-Large (MN). The dynamic CNNs consist of dynamic convolutions (Dy-Conv), dynamic ReLU (Dy-ReLU), and Coordinate Attention (CA) modules [13, 14, 15]. The Dy-Conv can extract noise-invariant features; Dy-ReLU increases the model's expressiveness by applying a dynamic non-linear function; and CA detects important channels, time frames, and frequency bins. These dynamic components are integrated into efficient inverted residual blocks in dynamic MN (DyMN) [16]. The DyMNs are defined using the parameter of *width_mult*, e.g., *DyMN10* is the DyMN with *width_mult* of 1.0.

Teacher Model	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6
Pre-trained DyMNs	*DyMN10*	*DyMN20*	*DyMN20*	*DyMN20*	*DyMN20*	*DyMN20*
Data Augmentation	FMix Fmask	FMix Fmask	Mixup Fmask	Mixup Fmask	FMix Fmask	FMix Fmask
Freq-MixStyle	✓	✓	✓	✓	✓	✓
HL in Teacher	×	×	OB	TB	OB	TB
split5	0.477	0.498	0.496	0.493	0.498	0.495
split10	0.523	0.547	0.543	0.542	0.553	0.545
split25	0.563	0.578	0.585	0.591	0.582	0.579
split50	0.585	0.603	0.616	0.620	0.605	0.599
split100	0.609	0.627	0.649	0.654	0.630	0.627

Table 1: The performance of each split for the teacher models. *DyMN10* and *DyMN20* are dynamic MoblieNets with different *width_mult*. OB and TB are one-branch and two-branch methods of hierarchical learning.

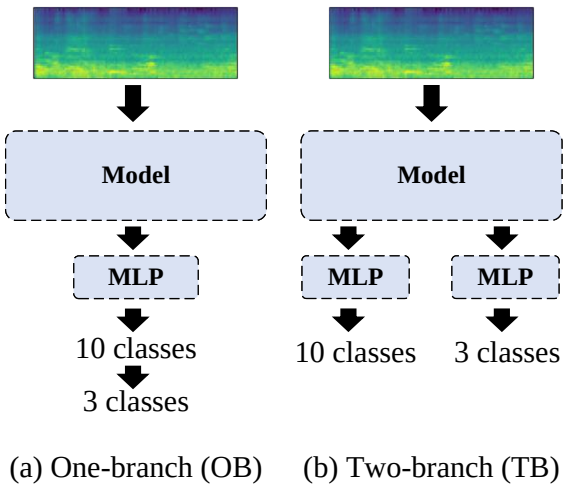

Figure 2: Two different architectures of hierarchical learning.

2.2.2. *CP-Mobile*

CP-Mobile was proposed as a novel efficient architecture for ASC [3]. CP-Mobile consists of residual inverted bottleneck blocks and incorporates different normalizations with the model. CP-Mobile maintains the representation capability of the CP-ResNet, making it efficient even in low-complexity settings.

2.3. Hierarchical learning

Some research introduced a hierarchical taxonomy for ASC, where the acoustic scene classes can be coarsely grouped into indoor, outdoor, and transportation (vehicle) [11, 12]. They used the hierarchical taxonomy as an auxiliary task to make the model learn coarse-grained and fine-grained acoustic information. Therefore, we propose 2 hierarchical learning methods. For the first method, we map the output of 10 acoustic scene classes into 3 coarse-grained classes and additionally calculate the cross-entropy loss of the 3-class classification. For the second method, we introduced a multi-task learning method using a two-branch network. The two methods are illustrated in Fig 2.

2.4. Quantization-aware training

QAT models the effects of quantization during training, allowing for higher accuracy compared to other quantization methods. QAT is introduced to further compress the CP-Mobile networks and meet the complexity requirements. During training, all calculations are performed in floating point, with fake quantization modules modeling the effects of quantization by clamping and rounding to simulate the effects of INT8 quantization. After model conversion, weights and activations are quantized, and activations are fused into the preceding layer where possible.

2.5. Data augmentation and device generalization methods

1 Mixup. It constructs a new training example by linearly interpolating two random examples from the training set and their labels[17].

2 FMix [18] can effectively augment the training data by randomly mixing irregular areas of two samples, and it is proven to be effective in improving the performance in ASC [19].

3 Frequency mask (Fmask). We randomly mask the frequency components of the spectrogram.

4 We apply Freq-MixStyle to address the device generalization problem [20].

3. EXPERIMENT

3.1. Experimental settings

We follow most of the baseline settings during feature extraction. We only change the *hop_length* to 505. We normalized each Mel-bin for training data and saved the mean and var values to normalize test and evaluation data.

We use the pre-trained *DyMN10* and *DyMN20* as the pre-trained dynamic CNNs. The *base_channels*, *channels_multiplier*, and *expansion_rate* of CP-Mobile are set to 32, 2.3, and 3, separately.

We use Adam optimizer with a learning rate of 1e-4 for fine-tuning the pre-trained DyMNs and QAT, while in KD we train the CP-Mobile with a learning rate of 1e-3. The batch size is set to 48.

3.2. Results

We fine-tuned *DyMN10* as the teacher model 1 without using HL. Then we fine-tuned *DyMN20* as the teacher model 2 to 6 with or with different HL and data augmentation methods. For all models, we implemented Fmask and Freq-MixStyle for all models to

Submission	System 1	System 2	System 3	Baseline
Ensemble Models	Model 1-2	Model 3-4	Model 1-6	
Student		CP-Mobile		
HL in Student	×	OB	×	
Data Augmentation	Fmask	Fmask	Fmask	
Freq-MixStyle	✓	✓	✓	
split5	0.426	0.429	0.422	0.424
split10	0.471	0.469	0.492	0.453
split25	0.540	0.517	0.541	0.503
split50	0.570	0.562	0.564	0.532
split100	0.594	0.593	0.603	0.570
Average	0.520	0.514	0.524	0.496
Parameters		126,952		122,296
MACS		28,893,268		29,419,156

Table 2: The details of performance on each split, parameters, and MACS for submitted systems. The performance is evaluated using the quantized CP-Mobile models.

improve the device generalization performance. Table 1 shows the performance of each split for the 6 teacher models.

From the table, comparing the performance of model 1 and 2, we can see that the fine-tuned *DyMN20* surpass *DyMN10* for each split. Comparing the performance of model 2 and 5, the performance is further improved, indicating the effectiveness of introducing the HL. When we use Mixup instead of FMix in HL, we achieve higher scores on *split25*, *split50*, *split100* but lower scores on *split5* and *split10*. This shows that FMix is more effective for smaller datasets. Comparing the results of 2 different HL methods, the two-branch method works when applying Mixup instead of FMix. Moreover, the proposed DyMNs with HL and data augmentation methods even surpass some ensembles of PASST or CP-ResNet on *split100*.

We developed 3 different systems for the final evaluation. We used the logits averaged from model 1 and 2 and trained the CP-Mobile for system 1. For system 2, we averaged from model 1 and 2 and trained the CP-Mobile with the one-branch HL method. For system 3, we averaged the 6 logits of models in Table 1 and trained the CP-Mobile without using HL method. Table 3.2 shows the performance of each split for the submitted systems. As shown in the table, all the 3 systems outperform the baseline with about 127K parameters and 29M MACS.

4. CONCLUSION

In this technical report, we propose a hierarchical learning-based ASC method with knowledge distillation and ensembles of dynamic CNNs. We introduced 2 different hierarchical learning methods with data augmentation methods when fine-tuning the pre-trained DyMNs. We average the logits of different teacher models and distill the CP-Mobile models with the knowledge from teacher models. We further use quantization-aware training to compress the student models. Experimental results show that our systems outperform the baseline for each split on the test set.

5. ACKNOWLEDGMENT

We acknowledge the support provided by the China Scholarship Council during a visit of Jisheng Bai to Nanyang Technological University.

6. REFERENCES

[1] T. Virtanen, M. D. Plumbley, and D. Ellis, *Computational analysis of sound scenes and events*. Springer, 2018.

[2] B. Ding, T. Zhang, C. Wang, G. Liu, J. Liang, R. Hu, Y. Wu, and D. Guo, "Acoustic scene classification: A comprehensive survey," *Expert Systems with Applications*, vol. 238, p. 121902, 2024.

[3] F. Schmid, T. Morocutti, S. Masoudian, K. Koutini, and G. Widmer, "CP-JKU submission to DCASE23: Efficient acoustic scene classification with Cp-Mobile," DCASE2023 Challenge, Tech. Rep., May 2023.

[4] M. Wang, R. Wang, B. Wang, J. Bai, C. Chen, Z. Fu, J. Chen, X. Zhang, and S. Rahardja, "Ciaic-ASC system for DCASE 2019 challenge task1," DCASE2019 Challenge, Tech. Rep., 2019.

[5] J. Bai, M. Wang, H. Liu, H. Yin, Y. Jia, S. Huang, Y. Du, D. Zhang, M. D. Plumbley, D. Shi, *et al.*, "Description on ieee icme 2024 grand challenge: Semi-supervised acoustic scene classification under domain shift," *arXiv preprint arXiv:2402.02694*, 2024.

[6] 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," *arXiv preprint arXiv:2405.10018*, 2024.

[7] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "PANNs: Large-Scale Pretrained Audio Neural Networks for Audio Pattern Recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.

[8] K. Chen, X. Du, B. Zhu, Z. Ma, T. Berg-Kirkpatrick, and S. Dubnov, "HTS-AT: A hierarchical token-semantic audio transformer for sound classification and detection," in *International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2022, pp. 646–650.

[9] F. Schmid, K. Koutini, and G. Widmer, "Dynamic convolutional neural networks as efficient pre-trained audio models," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2024.

[10] 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.

[11] Y. Xu, Q. Huang, W. Wang, and M. D. Plumbley, "Hierarchical learning for dnn-based acoustic scene classification," *arXiv preprint arXiv:1607.03682*, 2016.

[12] T. L. Nwe, T. H. Dat, and B. Ma, "Convolutional neural network with multi-task learning scheme for acoustic scene classification," in *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2017, pp. 1347–1350.

[13] Y. Chen, X. Dai, M. Liu, D. Chen, L. Yuan, and Z. Liu, "Dynamic convolution: Attention over convolution kernels," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 030–11 039.

[14] ——, "Dynamic relu," in *European Conference on Computer Vision*. Springer, 2020, pp. 351–367.

[15] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 13 713–13 722.

[16] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[17] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[18] E. Harris, A. Marcu, M. Painter, M. Niranjan, A. Prügel-Bennett, and J. Hare, "Fmix: Enhancing mixed sample data augmentation," *arXiv preprint arXiv:2002.12047*, 2020.

[19] J. Bai, J. Chen, M. Wang, M. S. Ayub, and Q. Yan, "A squeeze-and-excitation and transformer based cross-task model for environmental sound recognition," *IEEE Transactions on Cognitive and Developmental Systems*, 2022.

[20] K. Zhou, Y. Yang, Y. Qiao, and T. Xiang, "Domain generalization with mixstyle," *arXiv preprint arXiv:2104.02008*, 2021.