# MOFLENET: A LOW COMPLEXITY MODEL FOR ACOUSTIC SCENE CLASSIFICATION

*Oo Yifei*[1*†] *Nagisetty Srikanth*[2†] *Chong Soon Lim*[2]

[1]Nanyang Technological University, Singapore {yoo001@e.ntu.edu.sg}
[2]Panasonic R&D Center Singapore
{srikanth.nagisetty@sg.panasonic.com, chongsoon.lim@sg.panasonic.com}

## ABSTRACT

Designing lightweight models that require minimal computational resources and can operate on edge devices is the latest trend in deep learning research. This paper details our approach to Task 1: Low-Complexity Acoustic Scene Classification (ASC) for the DCASE'24 challenge. The task involves developing data-efficient systems for five scenarios, which progressively limit the available training data (i.e., 100%, 50%, 25%, 10%, 5%), while also handling device mismatches and low-complexity constraints (maximum memory allowance for model parameters: 128 kB, maximum number of MACs per inference: 30 million). In this work, we introduce a lightweight novel CNN architecture called MofleNet, featuring a combination of shuffle channels and residual inverted bottleneck blocks. Furthermore, we improve the performance by ensembling MofleNet with CP-ResNet. To meet the constraint of keeping the model size under 128 kB, both models are fine-tuned using quantization-aware training. Compared to the DCASE'24 Task 1 baseline, our proposed system improves results on the TAU Urban Acoustic Scenes 2022 Mobile Development dataset by around 6% on an average across five datasets and 4% on the challenge test set, earning a 7th rank in the DCASE'24 task 1 challenge.

***Index Terms***— MofleNet, CP-ResNet, Ensemble Learning, Quantization Aware Training, Device Impulse Response augmentation, Freq-MixStyle

## 1. INTRODUCTION

Acoustic Scene Classification (ASC) is a key research area within computational auditory scene analysis, focusing on categorizing audio recordings into predefined scene types. ASC has the potential to enhance various applications, including wearable devices, robotics, smart home devices, autonomous vehicles, and environmental monitoring. The annual *IEEE* DCASE Challenge has driven significant progress in ASC over the years.

In the *IEEE* DCASE'24 Challenge Task 1 [1], the goal is to classify 1-second audio recordings into one of ten predefined acoustic scene classes under three challenging conditions: (1) a recording device mismatch, (2) low complexity constraints, and (3) limited training data. For the training data, five scenarios with data subsets containing data approx. 5%, 10%, 25%, 50%, and 100% are provided. A system must only be trained on the specified subset and the explicitly allowed external resources. Additionally, to ensure ASC systems perform well on typical edge devices, strict constraints are imposed, limiting model size to 128 kB and multiply-accumulate operations (MACs) per inference to 30 million.

Convolutional Neural Networks (CNNs) dominates ASC tasks. Lightweight models like MobileNet variant CP-Mobile [4], GhostNet [5], SepNet [6] and blueprint separable convolutions network [7] have been used to tackle DCASE Task 1 challenges prior to 2023. In the DCASE'23 Task 1 challenge, the rank-1 model used an ensemble of 12 teacher models, including six variants of Patchout FaSt Spectrogram Transformer (PaSST) and six variants of CP-ResNet [3], to train a student model, CP-mobile (CPM) [4]. CPM's performance depends heavily on the number of channels in each CPM Block, but reducing the model size often requires sacrificing accuracy. Moreover, scaling down the model size does not proportionately decrease the MACs, presenting a significant challenge in balancing model size, accuracy, and computational efficiency. This year's challenge requires training the system on five different sizes of training sets. Training the teacher model on a 100% dataset to distill knowledge into a student model trained on a smaller dataset is not allowed. As a result, adopting the same approach as the top-ranked submission would require training 12 models for each of the five dataset sizes, totaling to 60 teacher models, making the process highly resource-intensive.

This paper introduces MofleNet (MobileShuffleNet), a model that incorporates channel shuffling and residual inverted bottleneck blocks into the CNN network. MofleNet is efficiently designed to meet the challenge requirements and address CP-Mobile's limitations. To further improve performance, we consider an ensemble of MofleNet and optimized CP-ResNet. The remainder of the paper is structured as follows: Section 2 discuss the data preprocess. Section 3 presents the MofleNet model. Section 4 covers ensemble models. The experimental setup is covered in Section 5. Section 6 discuss results, and finally, the conclusions are presented in Section 7.

## 2. DATA PRE-PROCESS

### 2.1. Dataset

The development dataset for this challenge is TAU22 [2], containing recordings from 12 European cities and capturing 10 distinct acoustic scenes using 4 real devices. Additionally, synthetic data for 11 mobile devices was generated based on the original recordings. TAU22 retains the content of the TAU Urban Acoustic Scenes 2020 Mobile development dataset (TAU20) but segments the 10- second audio clips into 1-second fragments, significantly increasing prediction difficulty. The dataset comprises

[†] Both authors contributed equally.
[*] Work was done during an internship at Panasonic R&D Center Singapore

230,350 1-sec audio clips, each labeled with corresponding acoustic scene. All audio clips are single-channel, 44.1 kHz and 24-bit format.

### 2.2. Feature Extraction

Raw 1D time domain audio signals were resampled to 32 kHz and converted to Mel domain. To obtain the Mel spectrogram, time domain signal is converted to the time frequency domain using short-time Fourier transform (STFT). This ensures that both the temporal and spectral characteristics of the audio data are utilized. After the frequency domain conversion, we extracted the Mel spectrogram corresponding to each audio clip using 256 Mel bands covering upto16 kHz. For the STFT Parameters, we employ a window size of 96 ms with a hop size of 16 ms for MofleNet and 23.4 ms as hop size for CP-ResNet. Input (features extracted) is arranged in the form of Frequency Bands X Time Frames X Channels.

### 2.3. Data Augmentations

To mitigate overfitting, especially for limited labelled data and to achieve good generalization, combination of Freq-MixStyle, Device Impulse Responses, and time rolling techniques are used.

**Frequency MixStyle (FMS)** is the frequency-wise version of MixStyle. It mixes frequency-wise statistics instead of channel-wise statistics in audio processing tasks [8]. MixStyle enhances model robustness to domain shifts by normalizing input features using the mean and standard deviation of other samples within the same batch, leveraging the observation that instance-wise statistical moments encapsulate style information. FMS normalizes the frequency bands in a spectrogram and then denormalizes them with mixed frequency statistics of two spectrograms. FMS is applied to a batch with a probability specified by the hyperparameter $p_{FMS}$, and the mixing coefficient is drawn from a Beta distribution of parameter α.

**Device Impulse Response (DIR)** augmentation involves convolving the input recordings with impulse responses from 66 freely available DIRs [9] from MicIRP [10]. The characteristic frequency responses of the recording devices in MicIRP make them ideal for simulating a diverse range of recording devices. This technique is designed to enhance the model's ability to generalize across recordings from various devices. DIR augmentation is controlled by the hyperparameter $p_{DIR}$, which defines the probability of convolving a waveform with a DIR.

**Time Rolling** involves shifting a prefix/suffix of a randomly sampled length to the other end of the input signal. This augmentation, computed in the time domain, helps to simulate variations in the temporal alignment of the audio data.

Following parametric values are used for data augmentation.

Table 1: Data Augmentation Parameters

| Model Name | FMS | | DIR | Time Rolling |
|---|---|---|---|---|
| | $p_{FMS}$ | α | $p_{DIR}$ | |
| **MofleNet** | 0.4 | 0.3 | 0.6 | 125ms |
| **CP-ResNet** | 0.8 | 0.4 | 0.4 | 125ms |

## 3. MOFLENET

Our proposed MofleNet architecture (MofleNet127) is depicted in Figure 1. It combines strided convolutions, Mofle Blocks, and average pooling to aggregate all components from the last convolution layer to obtain the scene prediction probabilities. The design of MofleNet was inspired from CP-Mobile [4] and ShuffleNet [11].

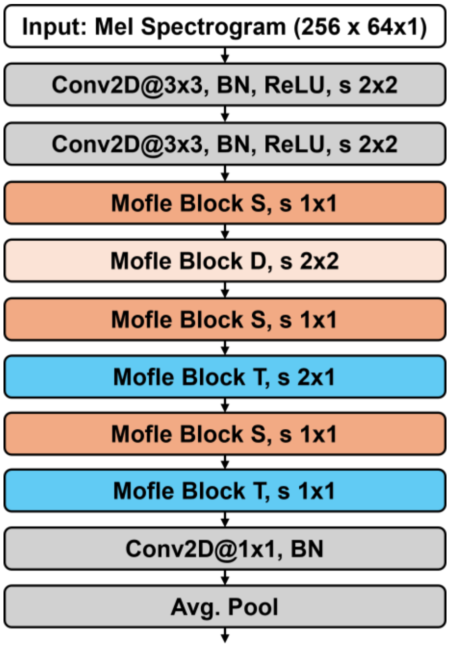

Figure 1: MofleNet127 Architecture:
Conv2D@KxK: Conv2D with Kernel Size KxK
s-Stride, BN-Batch Normalization

The Mofle block integrates grouped convolution, channel shuffle, depth wise convolution, and pointwise projection convolution to create a residual inverted bottleneck block. Figure 3 illustrates the S (Standard)/D (Spatial Down sampling) /T (Transition) design of Mofle Blocks. Unlike the CPM block, which employs pointwise expansion convolution, the Mofle Block replaces this with grouped convolution. A drawback of grouped convolution is that some channels outputs are derived from only a small fraction of input channels, limiting information exchange. To address this, channel shuffle (See Figure 2) was introduced after the grouped convolution to enhance information flow between channel groups. This promotes better mixing of information across different groups of channels, capturing more diverse and comprehensive features. This approach reduces the number of parameters and computational cost without significantly compromising information exchange between channels, resulting in richer and more informative feature maps. Additionally, the fourth layer of ShuffleNet units in [11] employs a grouped convolution, our experiments showed this configuration did not demonstrate significant improvement. Therefore, the Mofle block design doesn't include a grouped convolution layer after the depth wise convolutions.

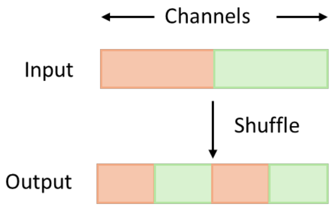

Figure 2: Channel Shuffle

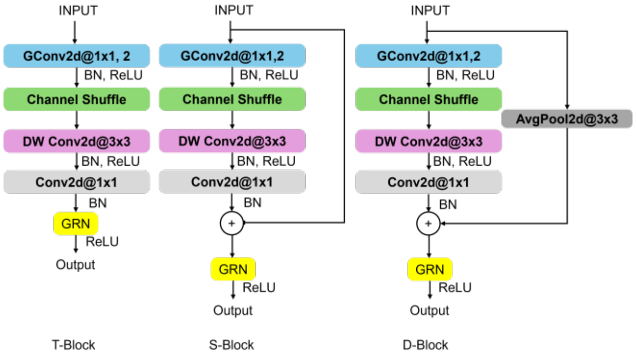

Figure 3: Mofle Blocks: T (Transition)-Block, S (Standard)-Block, D (Spatial Down sampling)-Block

**T-block**: The T-Block is designed to increase the number of channels within the network. These channels help in learning features across the 2D dimension, enabling the network to capture various patterns and representations from the input data.

**S-block**: The S-Block includes a residual connection, which helps mitigate vanishing gradient issues and facilitates the training of MofleNet. This block allows the model to learn both the original representation and the residual, leading to smoother optimization and better gradient flow.

**D-block**: The primary function of the D-Block is to reduce the model's complexity, particularly the MACs. It achieves this by decreasing the size of the feature maps, allowing the model to handle smaller data sizes more efficiently.

**Global Response Normalization (GRN)** [15] is applied before the final ReLU activation. GRN in Mofle blocks is used to avoid feature redundancies in models with restricted capacity.

## 4. ENSEMBLE MODELS

To enhance performance, we ensembled MofleNet and CP-ResNet after optimizing both models to meet challenge constraints. The resulting model sizes are 57kB for MofleNet127 (now referred as MofleNet57) and 59kB for CPR128 (now referred as CPR59), totaling 116kB.

### 4.1. MofleNet57

To lower the MACs without majorly impacting the accuracy, third Mofle Block in Figure 1 was tuned from Block S to Block D with a stride of (2x1) during convolution. Additionally, adjusted the channel multiplier and expansion rate to 1.8 and 2.6 respectively to further reduce model size and computations.

### 4.2. CPR59

CP-ResNet is a receptive-field regularized CNN that gradually builds local features covering a spatially restricted size. Table 2 presents the CPR59 architecture, a modified CP-ResNet, that ranked 1st in the DCASE'22 Task 1 challenge [3]. The original CP-ResNet model (CPR128) has approximately 128k parameters with a model size of 128kB. To reduce the model size and complexity, the following modifications were introduced:

- The number of parameters in the CP-ResNet network grows quadratically with its width. Reducing the channel multiplier from 2.0 to 1.4 brings down the parameter count below 64,000 (50% reduction in model size).
- Introducing max pooling layers with a shape of (2x1) and stride of (2x1) before the third and fourth (also the last) blocks reduces the MACs to under 15 million.

For additional information on the Basic CPR block listed in Table 2, refer to Figure 4.

Table 2: CPR59 Architecture

| Operator | Output Shape |
|---|---|
| Input | 256 x 43 x 1 |
| Conv2D@3x3, BN, ReLU | 127 x 21 x 32 |
| Max Pool | 63 x 10 x 32 |
| Basic CPR Block(A) | 63 x 10 x 32 |
| Max Pool | 31 x 10 x 32 |
| Basic CPR Block(A) | 31 x 10 x 32 |
| Max Pool | 15 x 10 x 32 |
| Max Pool | 7 x 10 x 32 |
| Basic CPR Block(B) | 7 x 10 x 44 |
| Max Pool | 7 x 5 x 44 |
| Basic CPR Block(B) | 7 x 5 x 26 |
| Conv2D@1x1, BN | 7 x 5 x 10 |
| Avg. Pool | 1 x 1 x 10 |

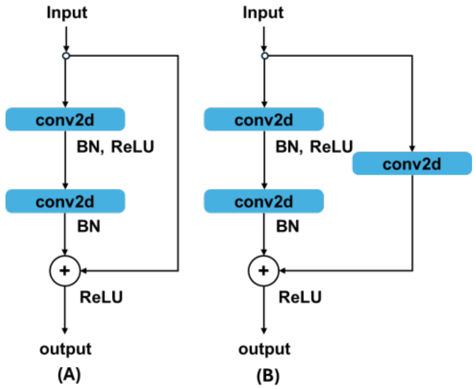

Figure 4: Two Basic CPR Blocks

## 5. EXPERIMENTAL SETUP

### 5.1. Training:

A total of 150 epochs with a batch size of 256, and adam optimizer was used for training. The learning rate strategy follows the same approach as in [4].

### 5.2. Quantization Aware Training:

As a part of Task 1 challenge constraints, submitted models should meet 128kB as memory requirement. To minimize the drop in performance after the quantization step, we applied Quantization Aware Training (QAT) [13] to all our architectures by fine-tuning the models for 24 epochs. A peak learning rate of 5×10−5 and linearly decreased it to 10% by epoch 16 was set during fine-tuning phase. Conv2d + BN + ReLU combinations was fused into a single layer and utilized PyTorch's 'fbgemm' quantization configuration [12]. All computations were performed in int8, except for those in the GRN layer of MofleNet. Table 3 presents the total parameters, model size and Million MACs (MMAC) per inference.

Table 5: DCASE'24 Top Submission Results

| Submission Label | Rank | Accuracies per Split | | | | | Key Contribution | Knowledge Distillation |
|---|---|---|---|---|---|---|---|---|
| | | 100% | 50% | 25% | 10% | 5% | | |
| **Han_SJTUTHU** | 1 | 61.82 | 60.38 | 59.09 | 56.69 | 54.35 | Model Pruning | 4 Teachers models |
| **MALACH24_JKU** | 2 | 61.51 | 60.05 | 58.01 | 54.46 | 51.95 | New training strategy | 3 Teacher models with Bayesian Ensemble |
| **Shao_NEPUMSE** | 3 | 61.71 | 60.61 | 58.31 | 53.75 | 51.38 | Mamba variation | 12 Teacher models |
| **OO_NTUPRDCSG** | 7 | 59.91 | 58.42 | 55.87 | 51.43 | 48.52 | MofleNet | Not Utilized |

Table 3: Total Parameters, Model Size and Complexity

| Model | Parameters | Size (kB) | MMAC |
|---|---|---|---|
| **Baseline** | 61,000 | 122 | 29 |
| **MofleNet127** | 127,000 | 127 | 27.7 |
| **MofleNet57** | 57,000 | 57 | 13.4 |
| **CPR59** | 59,000 | 59 | 16 |
| **MofleNet57+CPR59** | 116,000 | 116 | 29.4 |

## 6. RESULTS

### 6.1. Development Results

The performance of the models for the five scenarios (100%, 50%, 25%, 10%, 5%) on the validation data is shown in Table 4, the data splits are predefined by the challenge organizers. On average, the MofleNet127 and CPR128 architectures demonstrate a 4% performance improvement compared to the baseline. Notably, MofleNet127 performed well on the 100%, 50%, 25%, and 10% datasets but shows limited improvement on the 5% dataset. In contrast, CPR128 [3] outperforms MofleNet127 on the 5% dataset by 4.1%.

Table 4: Model accuracies after QAT

| Model | Accuracies per Split | | | | |
|---|---|---|---|---|---|
| | 100% | 50% | 25% | 10% | 5% |
| **Baseline** | 56.99 | 53.19 | 50.29 | 45.29 | 42.40 |
| **MofleNet127** | 61.94 | 58.68 | 55.4 | 49.1 | 42.94 |
| **CPR128** | 60.06 | 58.88 | 55.18 | 50.82 | 47.08 |
| **MofleNet57** | 58.79 | 56.71 | 52.21 | 45.4 | 41.22 |
| **CPR59** | 58.49 | 57.52 | 54.81 | 48.79 | 44.92 |
| **MofleNet57+ CPR59** | **62.22** | **60.04** | **56.73** | **51.27** | **47.59** |

Using MofleNet57 or CPR59 individually, without ensembling, yields only marginal improvements over the baseline model, whereas the ensemble approach achieves significantly better results. Although the individual performance of MofleNet57 and CPR59 models is notable, their true value lies in the significant savings on MACs and model size.

Development results demonstrate that the ensemble of the two models significantly improves accuracy by approximately 6% compared to the baseline.

### 6.2. Challenge Results

This section provides a critical analysis of the challenge results and submitted systems. Table 5 [14] displays the top team's submissions and rankings, with our team (OO_NTUPRDCSG) securing 7th place. Notably, our ensemble of MofleNet with CP-ResNet demonstrates a robust strategy, yielding a 4% performance increase on the challenge test data compared to the baseline, while reducing the model size from 122 kB to 116kB. Unlike the top models, which employed Knowledge Distillation and external data, our model was trained directly on development dataset subsets, showcasing its effectiveness in handling limited data without relying on additional resources.

Analysis of the top-ranked models revealed that both the 1st and 2nd rank submissions were fine-tuned versions of CP-Mobile [4]. Replacing 3x3 convolutions with a combination of 1x3 and 3x1 convolutions reduced CP-Mobile model complexity but did not improve performance, indicating that model pruning was the key contributor to the top-ranked submission's success. The 2nd place submission's key contribution lies in its novel training strategy for CP-Mobile.

Our approach closely aligns with the 3rd-ranked submission, with the primary difference being their use of Knowledge Distillation. MofleNet and CP-ResNet ensemble achieved 64% accuracy, nearing the 3rd-ranked submissions on the development dataset with Knowledge Distillation. Our key contribution is the development and strong performance of MofleNet and its ensemble.

## 7. CONCLUSIONS

In this work, we presented our approach for Task 1: Low-Complexity Acoustic Scene Classification in the DCASE 2024 challenge. We introduced MofleNet, a novel hybrid architecture incorporating shuffle channels and residual inverted bottleneck blocks and used it in an ensemble with CP-ResNet. Our methods included augmentation techniques such as Freq-MixStyle and Device Impulse Response, along with Quantization Aware Training to meet the model size constraint. Our experimental results demonstrated that the ensemble of MofleNet and CP-ResNet significantly improved accuracy compared to individual models by approx. 4% and baseline by approx. 6%. Specifically, MofleNet performed better with larger datasets, while CPR59 was more effective with smaller datasets. Additionally, DCASE'24 Task 1 challenge results demonstrate the strength and potential of our ensemble approach. Despite not utilizing Knowledge Distillation, our model demonstrated good performance in handling limited data scenarios. This work highlights the importance of model ensemble and novel design of MofleNet, setting a foundation for future advancements in this domain.

## 8. REFERENCES

[1] 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.

[2] T. Heittola, A. Mesaros, and T. Virtanen, "Acoustic scene classification in dcase 2020 challenge: Generalization across devices and low complexity solutions," in DCASE Workshop, 2020.

[3] F. Schmid, S. Masoudian, K. Koutini, and G. Widmer, " Knowledge distillation from transformers for low complexity acoustic scene classification," in DCASE Workshop, 2022.

[4] F. Schmid , T. Morocutti , S. Masoudian , K. Koutini , G. Widmer, "CP-JKU Submission to DCASE23: Efficient Acoustic Scene Classification with CP-Mobile," in DCASE, 2023.

[5] T. S. Kim, D. Rho, G. Lee, and J. H. Park, "Dual-Strategy Enhancement of Acoustic Scene and Event Classification: Integrating Res2Net, GhostNet, and MobileFormer Architectures," in DCASE, 2023

[6] Y. Cai, M. Lin, C. Zhu, S. Li, and X. Shao, "DCASE2023 Task 1 Submission: Device Simulation and Time-Frequency Separable Convolution for Acoustic Scene Classification," in DCASE, 2023.

[7] J. Tan, and Y. Li, "Low-Complexity Acoustic Scene Classification Using Blueprint Separable Convolution and Knowledge Distillation," in DCASE, 2023.

[8] 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," arXiv preprint arXiv:2206.12513, 2022.

[9] T. Morocutti, F. Schmid, K. Koutini, and G. Widmer, "Device-Robust Acoustic Scene Classification via Impulse Response Augmentation," in 2023 31st European Signal Processing Conference (EUSIPCO), pp. 176-180. IEEE, 2023.

[10] "Microphone Impulse Response Project," 2017. URL: https://micirp.blogspot.com/?m=1.

[11] X. Zhang, X. Zhou, M. Lin, J. Sun, "ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 6848-6856, 2018.

[12] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library, in Advances in Neural Information Processing Systems (NeurIPS)," advances in neural information processing systems 32, 2019.

[13] B. Jacob, S. Kligys, B. Chen, M. Zhu, M. Tang, A. Howard, H. Adam, and D. Kalenichenko, "Quantization and Training of Neural Networks for Efficient Integer-Arithmetic-Only Inference," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2704-2713, 2018.

[14] "Data-Efficient Low-Complexity Acoustic Scene Classification 2024 Task 1 Challenge Results, 2024. URL: https://dcase.community/challenge2024/task-data-efficient-low-complexity-acoustic-scene-classification-results

[15] S. Woo, S. Debnath, R. Hu, X. Chen, Z. Liu, I. S. Kweon, and S. Xie, "Convnext V2: co-designing and scaling convnets with masked autoencoders," CoRR, vol. abs/2301.00808, 2023.