DCASE 2021 Task 1A: Technique Report

AIT Group: Lam Pham, Alexander Schindler, Anahid Jalali, Hieu Tang, Hoang Truong
Competence Unit Data Science & Artificial Intelligence,
Center for Digital Safety & Security,
Austrian Institute of Technology, Austria.

***Abstract*—In this report, we presents a low-complexity deep learning frameworks for acoustic scene classification (ASC). The proposed framework can be separated into three main steps: Front-end spectrogram extraction, back-end classification, and late fusion of predicted probabilities. In the first step, we use Mel filter, Gammatone filter and Constant Q Transfrom (CQT) to transform draw audio signal into spectrograms. Three spectrograms are then feed into three individual back-end convolutional neural networks (CNNs) for classification. Finally, a late fusion of three predicted probabilities obtained from three CNNs is conducted to achieve the final classification result. To reduce the complexity of CNN network architecture proposed, we apply two model compression techniques: model restriction and decomposed convolution. Our experiments, which are conducted on DCASE 2021 Task 1A development dataset, achieve a low-complexity CNN based framework with 128 KB trainable parameters and the best classification accuracy of 66.7%, improving DCASE baseline by 19.0%.**

***Index Terms*—Convlutional neural network, Gammatone filter, constant Q transoform, MEL filter, spectrogram, deep learning.**

I. Introduction

To deal with ASC challenges such as unbalanced data, lacking of data input, or mismatch recording devices, various methods have been proposed, which can be separated into two main approaches. The first approach makes use of multiple data input such as ensemble of spectrograms [1]–[3] or audio channels [4]. Meanwhile, the second approach focuses on back-end classification, proposes powerful deep learning network architectures which are able to enforce the training process [5]–[8]. Although these two approaches can achieve good results, they present high-complexity systems. Indeed, while multiple input data requires an ensemble of multiple individual classification models [9], [10], powerful network architectures show a number of convolutional layers [5], [6]. All top-10 systems proposed in recent DCASE chalenges in 2018, 2019, 2020 also show large architectures, requiring larger than 2 MB of trainable parameters. The issue of network complexity prevents applying for edge devices with respect to real-life applications. Although there are various methods proposed to deal with the issue of model complexity such as quantizaton [11], pruning [12], [13], model restriction (i.e. restriction on the number of layers [14] or the number of kernel [15] or both of these factors [16]), decomposed convolution [17], or hybrid methods using pruning and decomposed convolution [17], pruning and distillation [18], these are mainly applied for image data. This report therefore introduces a low-complexity deep learning framework for ASC. To deal with ASC challenges, we propose an ensemble of multiple spectrogram inputs, using Mel filter [19], Gammaton [20] filter, and CQT [19]. For each network used for training an individual spectrogram input, we deal with the issue of model complexity by combining model restriction and decomposed convolution methods.

II. The low-complexity deep learning framework proposed

A. Our baseline

To evaluate, we firstly propose a baseline with high-level architecture shown in Fig. 1. Initially, a draw audio signal is firstly transformed into a spectrogram by using MEL filter [19]. Next, the spectrogram are split into patches before feeding into a CNN based network for classification. As the CNN based network architecture proposed is shown in Table I, it contains sub-blocks which perform convolution with C_{out} channel (Convolution ([kernel size]@C_{out})), batch normalization (BN) [21], rectified linear units (ReLU) [22], average pooling (AV [kernel size]), global average pooling (GAP), dropout (percentage dropped) [23], fully-connected (FC), and Softmax layers. In total, we have 6 convolutional layers and 1 fully-connected layers that makes the proposed network architecture like CNN-7. As the CNN-7 works on patches, the final predicted probability of an entire spectrogram is computed by averaging of all patches. As we use three spectrogram input (CQT, log-mel, and Gammatonegram) as a rule of thumb to improve ASC performance [9], [10], an ensemble of these predicted probabilities obtained from three spectrogram inputs is applied.

B. Model compression methods applied to the CNN-7 network

Regarding the CNN-7 architecture proposed, it reports a complexity of 1,129 MB for non-zero parameters with using 32 bits for representing one trainable parameter. Additionally, using ensemble of three spectrogram inputs make the the number of trainable parameters further increase three times. To reduce the model complexity, we firstly restrict the number of channels used in the CNN-7 baseline, then reduce the channels of $C_{out}1$ from 32 to 16, $C_{out}3$ and $C_{out}4$ from 64 to 32, $C_{out}5$ and $C_{out}6$ from 128 to 64. The channel restriction (CR) proposed helps to reduce the CNN-7 complexity to 313 KB that nearly equals to 1/4 of the original size.

We further reduce the CNN-7 complexity by applying the decomposed convolution (DC) technique described in [17], [24]. Let us consider C_{in} and C_{cout} as the input and output channel numbers respectively, $W = 3$ and $L = 3$ are the

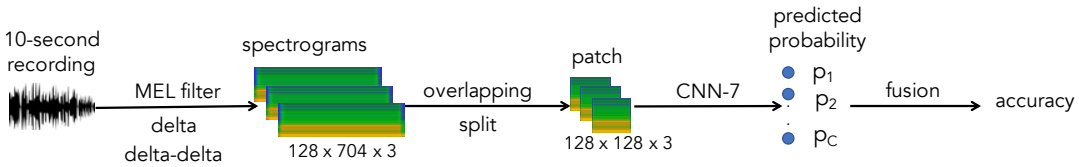

Fig. 1. High-level architecture of ASC baseline system proposed.

TABLE I
THE CNN-7 NETWORK ARCHITECTURE BASELINE (INPUT PATCH OF 128×128×3)

Network architecture	**Output**
BN - Convolution ([3×3]@$C_{out}1 = 32$) - ReLU - BN - Dropout (10%)	128×128×32
BN - Convolution ([3×3]@$C_{out}2 = 32$) - ReLU - BN - AP [2×2] - Dropout (10%)	64×64×32
BN - Convolution ([3×3]@$C_{out}3 = 64$) - ReLU - BN - Dropout (10%)	64×64×64
BN - Convolution ([3×3]@$C_{out}4 = 64$) - ReLU - BN - AP [2×2] - Dropout (10%)	32×32×64
BN - Convolution ([3×3]@$C_{out}5 = 128$) - ReLU - BN - AP [2×2] - Dropout (10%)	16×16×128
BN - Convolution ([3×3]@$C_{out}6 = 128$) - ReLU - BN - GAP - Dropout (10%)	128
FC - Softmax	$C = 10$

dimensions of kernel size, which are used for a convolutional layer. Then, the number of trainable parameters at a convolutional layer is computed by $W \times L \times C_{in} \times C_{out} = 9 \times C_{in} \times C_{out}$. We reduce the number of trainable parameters at a convolutional layer by decomposing the convolutional layer into 4 sub-convolutional layers as described in Fig. 2. For all four sub-convolutional layers, the output channel is reduced to $C_{out}/4$. Regarding the first sub-convolutional layer (the upper path shown in Fig. 2), although we still use kernel size of [W×L]=[3×3], we reduce the input channels to C_{in}/4, then cost $(9 \times C_{in} \times C_{out})/16$ trainable parameters. Regarding the other sub-convolution layers, we reduce the kernel size to [W×L]=[1×1]. While the input channel is reduced to $C_{in}/2$ at the second and third sub-convolutional layers (two middle paths shown in Fig. 2), it is remained in the fourth sub-convolutional layer (the lower path shown in Fig. 2). As a result, it requires $(C_{in} \times C_{out})/8$ for the second and third sub-convolutional layers, and $(C_{in} \times C_{out})/4$ for the fourth sub-convolution layer. By decomposing a convolutional layer into four sub-convolutional layers, the model complexity is reduced to nearly 1/8.5 of the original size. By combining the two model compression techniques, we can achieve a CNN-7 network architecture with complexity of 42.6 KB, which nearly equals to 1/25 of the original size (i.e. the CNN-7 network architecture proposed in the baseline framework in Table I). As we need to use three CNN-7 for three different spectrogram inputs, the final complexity of the framework proposed is approximately 128 KB.

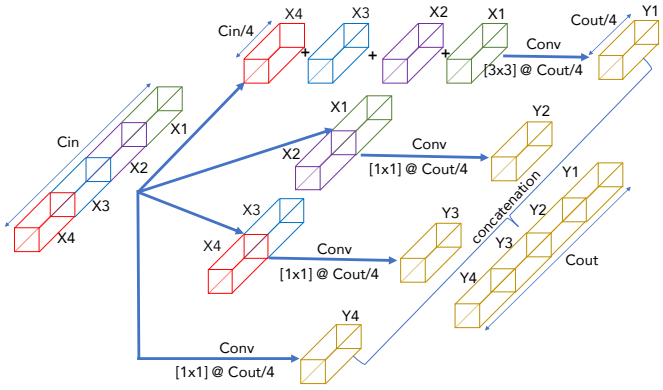

Fig. 2. Decomposed convolution technique applied to a convolutional layer.

III. Evaluation Setting

A. TAU Urban Acoustic Scenes 2020 Mobile, development dataset [25]

This dataset is referred to as DCASE 2021 Task 1A Development, which was proposed for DCASE 2021 challenge [26]. In this challenge, the limitation of model complexity is set to 128 KB with using 32 bits for one trainable parameter. The dataset in slightly unbalanced, recorded from 12 large European cities: Amsterdam, Barcelona, Helsinki, Lisbon, London, Lyon, Madrid, Milan, Prague, Paris, Stockholm, and Vienna. It consists of 10 scene classes: airport, shopping mall (indoor), metro station (underground), pedestrian street, public square, street (traffic), traveling by tram, bus and metro (underground), and urban park. The audio recordings were recorded from 3 different devices namely A (10215 recordings), B (749 recordings), C (748 recordings). Additionally, synthetic data for 11 mobile devices was created based on the original recordings, referred to as S1 (750 recordings), S2 (750 recordings), S3 (750 recordings), S4 (750 recordings), S5 (750 recordings), and S6 (750 recordings). As a result, this task not only requires low complexity model, but it also proposes an issue of mismatch device. To evaluate, we follow the DCASE 2021 Task 1A challenge [26], separate this dataset into training (Train.) and evaluation (Eval.) subsets. Then, Train. subset is used for training frameworks proposed and Eval. subset is used for evaluating. Notably, two of 12 cities and S4, S5, S6 audio recordings are only presented in the Eval. subset for evaluating the issue of mismatch recording devices and unseen samples.

TABLE II
PERFORMANCE COMPARISON AMONG DCASE BASELINE, THE CNN-7 BASELINE, THE CNN-7 BASELINE WITH CHANNEL RESTRICTION (CNN-7 W/ CR), AND THE CNN-7 BASELINE WITH CHANNEL RESTRICTION AND DECOMPOSED CONVOLUTION (CNN-7 W/ CR & DC).

Category	DCASE baseline (90.3 KB)	CNN-7 baseline (1.1 MB)	CNN-7 w/ CR (313 KB)	CNN-7 w/ CR & DC (42.6 KB)
Airport	40.5	59.5	50.3	64.5
Bus	47.1	73.7	70.4	69.0
Metro	51.9	57.6	49.8	70.0
Metro station	28.3	53.9	48.1	45.1
Park	69.0	73.1	78.5	74.4
Public square	25.3	34.3	38.4	25.9
Shopping mall	61.3	52.9	50.2	43.4
Street pedestrian	38.7	39.4	35.0	32.7
Street traffic	62.0	84.5	88.2	89.6
Tram	53.0	67.9	62.5	52.7
Average	47.7	59.7	57.1	56.7

TABLE III
THE NUMBER OF 10-SECOND AUDIO-VISUAL SCENE RECORDINGS CORRESPONDING TO EACH DEVICE IN THE TRAIN. AND EVAL. SUBSETS SEPARATED FROM THE DCASE 2021 TASK 1A DEVELOPMENT DATASET [27] AND PERFORMANCE FOR EACH DEVICES.

Devices	Train.	Eval.	Acc. %
A	10215	330	79.1
B	749	329	69.6
C	748	329	70.8
S1	750	330	65.8
S2	750	330	63.6
S3	750	330	67.0
S4	0	330	63.9
S5	0	330	60.0
S6	0	330	60.3

B. Deep learning framework implementation

We use Tensorflow framework to build all classification models in this papers. The cross-entropy loss function is used for training and Adam algorithm is used for optimization.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

A. Performance comparison between DCASE baseline and the CNN-7 baseline with or without using model compression methods

As experimental results are shown in Table II, although the CNN-7 baseline outperforms DCASE baseline and helps to improve the accuracy by 12%, the CNN-7 baseline complexity is much larger than DCASE baseline. By using model compression methods (CR & DC), we can achieve a low-complexity model referred to as CNN-7 with CR & DC, which is nearly 1/2 of the DCASE baseline complexity, but still outperforms DCASE baseline and achieves an accuracy improvement of 9%.

B. Evaluate ensemble of different spectrogram inputs

Given the optimized framework (CNN-7 with CR & DC), we conduct a fusion of three predicted probabilities from three spectrogram inputs to obtain the final classification accuracy. We then compare performances among DCASE baseline, the optimized framework with individual spectrograms, the optimized framework and ensemble of multiple spectrograms, across all scene categories. As experimental results are shown in Fig.3, GAM and MEL achieve competitive results, and outperform CQT at almost scene categories except for 'Airport' and 'Bus'. The ensemble of three spectrogram inputs helps to achieve an average accuracy of 66.7%, improving DCASE baseline by 19%, and notably showing improvement over all scene categories.

Further analysing performance over different recording devices as shown in Table III, we can see that device A outperforms the other devices as this device is dominant in Train. subset. Although there is a lacking of training samples for device B and C, they achieves competitive accuracy of 69.6% and 70.8% respectively, compared with device A performance of 79.1%. Regarding synthesized devices from S1 to S6, although there is no samples from S4, S5, S6 in Train. subset, the performance of these devices are competitive to the other S1, S2, S3. The analysis proves that the ASC framework proposed not only shows low complexity of 128 KB, it also can tackle the issue of mismatched recording devices.

V. CONCLUSION

We have just presented a low-complexity framework for ASC, which makes use multiple spectrogram inputs and model compression techniques. While the ensemble of multiple spectrograms helps to tackle different ASC challenges of mismatch recording devices or lacking of input to improve the performance, a combination of model restriction and decomposed convolution techniques is effective to achieve a low model complexity of 128 KB.

REFERENCES

[1] L. Pham, I. Mcloughlin, H. Phan, R. Palaniappan, and A. Mertins, "Deep feature embedding and hierarchical classification for audio scene classification," in *International Joint Conference on Neural Networks (IJCNN)*, 2020, pp. 1–7.

[2] D. Ngo, H. Hoang, A. Nguyen, T. Ly, and L. Pham, "Sound context classification basing on join learning model and multi-spectrogram features," *ArXiv*, vol. abs/2005.12779, 2020.

[3] T. Nguyen and F. Pernkopf, "Acoustic scene classification using a convolutional neural network ensemble and nearest neighbor filters," in *Proc. DCASE*, 2018, pp. 34–38.

[4] Y. Sakashita and M. Aono, "Acoustic scene classification by ensemble of spectrograms based on adaptive temporal divisions," DCASE Challenge, Tech. Rep., 2018.

[5] L. Pham, H. Phan, T. Nguyen, R. Palaniappan, A. Mertins, and I. Mcloughlin, "Robust acoustic scene classification using a multi-spectrogram encoder-decoder framework," *Digital Signal Processing*, vol. 110, p. 102943, 2021.

[6] S. Phaye, E. Benetos, and Y. Wang, "SubSpectralNet using sub-spectrogram based convolutional neural networks for acoustic scene classification," in *Proc. ICASSP*, 2019, pp. 825–829.

[7] Z. Ren, K. Qian, Y. Wang, Z. Zhang, V. Pandit, A. Baird, and B. Schuller, "Deep scalogram representations for acoustic scene classification," *IEEE/CAA Journal of Automatica Sinica*, vol. 5, no. 3, pp. 662–669, 2018.

[8] H. Song, J. Han, S. Deng, and Z. Du, "Acoustic scene classification by implicitly identifying distinct sound events," in *Proc. INTERSPEECH*, 2019, pp. 3860–3864.

[9] L. Pham, I. McLoughlin, H. Phan, R. Palaniappan, and Y. Lang, "Bag-of-features models based on C-DNN network for acoustic scene classification," in *Proc. AES*, 2019.

[10] L. Pham, I. Mcloughlin, H. Phan, and R. Palaniappan, "A robust framework for acoustic scene classification," in *Proc. INTERSPEECH*, 2019, pp. 3634–3638.

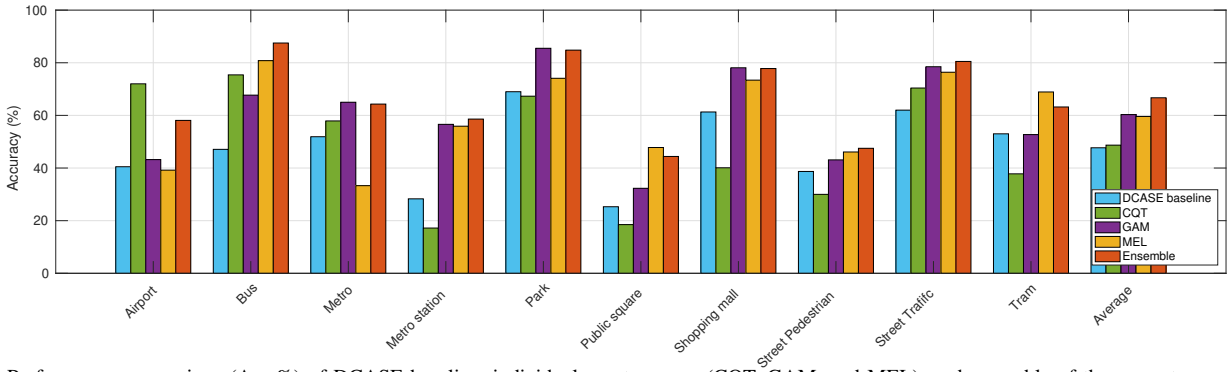

Fig. 3. Performance comparison (Acc.%) of DCASE baseline, individual spectrograms (CQT, GAM, and MEL), and ensemble of three spectrograms across all scene categories (using CNN-7 with CR & DC)

[11] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proceedings of the IEEE*, vol. 105, no. 12, pp. 2295–2329, 2017.

[12] S. Han, J. Pool, J. Tran, and W. J. Dally, "Learning both weights and connections for efficient neural networks," *arXiv preprint arXiv:1506.02626*, 2015.

[13] J. Frankle and M. Carbin, "The lottery ticket hypothesis: Finding sparse, trainable neural networks," *arXiv preprint arXiv:1803.03635*, 2018.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[15] S. Zagoruyko and N. Komodakis, "Wide residual networks," *arXiv preprint arXiv:1605.07146*, 2016.

[16] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*. PMLR, 2019, pp. 6105–6114.

[17] Y.-D. Kim, E. Park, S. Yoo, T. Choi, L. Yang, and D. Shin, "Compression of deep convolutional neural networks for fast and low power mobile applications," *arXiv preprint arXiv:1511.06530*, 2015.

[18] V. Joseph, S. A. Siddiqui, A. Bhaskara, G. Gopalakrishnan, S. Muralidharan, M. Garland, S. Ahmed, and A. Dengel, "Reliable model compression via label-preservation-aware loss functions," *arXiv preprint arXiv:2012.01604*, 2020.

[19] B. McFee, R. Colin, L. Dawen, D. Ellis, M. Matt, B. Eric, and N. Oriol, "librosa: Audio and music signal analysis in python," in *Proceedings of The 14th Python in Science Conference*, 2015, pp. 18–25.

[20] D. P. W. . Ellis, "Gammatone-like spectrogram," 2009. [Online]. Available: http://www.ee.columbia.edu/ dpwe/resources/matlab/ gammatonegram

[21] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning*, 2015, pp. 448–456.

[22] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *ICML*, 2010.

[23] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[24] K. Koutini, F. Henkel, H. Eghbal-zadeh, and G. Widmer, "Low-complexity models for acoustic scene classification based on receptive field regularization and frequency damping," *arXiv preprint arXiv:2011.02955*, 2020.

[25] A. Mesaros, T. Heittola, and T. Virtanen, "A multi-device dataset for urban acoustic scene classification," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events (DCASE)*, 2018, pp. 9–13.

[26] Detection and Classification of Acoustic Scenes and Events Community, *DCASE 2021 challenges*, http://dcase.community/challenge2021.

[27] S. Wang, A. Mesaros, T. Heittola, and T. Virtanen, "A curated dataset of urban scenes for audio-visual scene analysis," *arXiv preprint arXiv:2011.00030*, 2020.