EMNAPE: Efficient Multi-Dimensional Neural Architecture Pruning for EdgeAI

Hao Kong[1,2], Xiangzhong Luo[1], Shuo Huai[1,2], Di Liu[3], Ravi Subramaniam[4], Christian Makaya[4], Qian Lin[4], and Weichen Liu[1,*]
[1]School of Computer Science and Engineering, Nanyang Technological University, Singapore
[2]HP-NTU Digital Manufacturing Corporate Lab, Nanyang Technological University, Singapore
[3]Department of Computer Science, Norwegian University of Science and Technology, Trondheim, Norway
[4]HP Inc., Palo Alto, California, USA

***Abstract*—In this paper, we propose a multi-dimensional pruning framework, EMNAPE, to jointly prune the three dimensions (depth, width, and resolution) of convolutional neural networks (CNNs) for better execution efficiency on embedded hardware. In EMNAPE, we introduce a two-stage evaluation strategy to evaluate the importance of each pruning unit and identify the computational redundancy in the three dimensions. Based on the evaluation strategy, we further present a heuristic pruning algorithm to progressively prune redundant units from the three dimensions for better accuracy and efficiency. Experiments demonstrate the superiority of EMNAPE over existing methods.**

I. Introduction

Deep convolutional neural networks (CNNs) have achieved promising accuracy in various vision applications, such as image classification [1] and object detection [2]. However, better accuracy comes at the cost of higher computational complexity, which impedes the deployment of advanced CNNs onto resource-constrained embedded devices. To facilitate the deployment of CNNs onto embedded devices and advance the development of EdgeAI, model pruning has aroused widespread attention in reducing the computational redundancy in the three dimensions (depth, width, and resolution) of CNNs [3]–[5]. Nevertheless, existing pruning approaches mainly focus on reducing the redundancy in a single dimension, which ignores the redundancy in the other dimensions and thus only achieves very limited improvements in model efficiency.

To pursue higher execution efficiency without sacrificing accuracy, in this paper, we propose a multi-dimensional pruning framework, EMNAPE, to coordinately prune the three dimensions of CNNs. The main insights of the proposed pruning framework are a two-stage importance evaluation strategy and a heuristic pruning algorithm. Specifically, the two-stage evaluation strategy will conduct both inner-dimensional and inter-dimensional evaluation to determine the importance of each unit in terms of model computation (Multiply-Accumulate Operations, MACs), accuracy, and inference latency, according to which we are able to comprehensively identify redundant units in the three dimensions. Based on the proposed evaluation strategy, our heuristic pruning algorithm will progressively remove less sensitive units and obtain the optimal tiny model. By this means, we can comprehensively reduce the computational redundancy in all three dimensions of CNNs, thereby achieving higher execution efficiency on embedded devices without compromising accuracy.

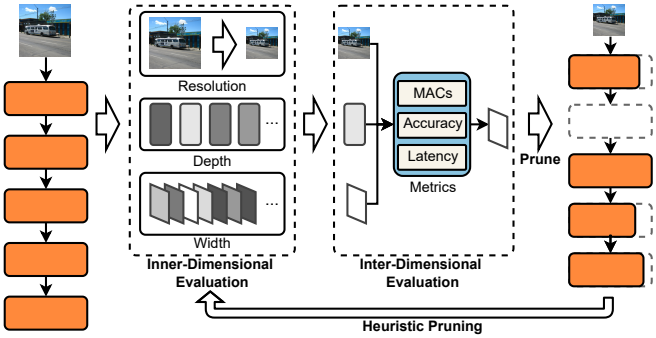

Fig. 1: The overview of EMNAPE. In inner-dimensional evaluation, the deeper the color, the more important the unit.

II. Multi-Dimensional Pruning

The architecture of the proposed framework is demonstrated in Fig. 1. Given an overparameterized CNN model, we first separately evaluate and compare the importance of each pruning unit within each dimension. Inspired by single-dimensional pruning [6], in this stage, we utilize the gradient of each unit as the importance metric, which can be represented as follows:

$$I_{in} = \left(\frac{\partial L}{\partial u}\right)^2 \tag{1}$$

where u denotes the pruning unit, L is the prediction loss, and I_{in} is the importance score of u within the corresponding dimension. According to Equation 1, we can efficiently evaluate the importance of units within each dimension and identify the most redundant unit of each dimension. To make the final pruning decision, we need to further compare the selected redundant unit of each dimension. However, the inner-dimensional importance metric only considers the impact of units on accuracy, which is unable to compare units of different dimensions since pruning units of different dimensions will lead to diverse impacts on model computation, accuracy, and

*Corresponding author
This study is partially supported under the RIE2020 Industry Alignment Fund – Industry Collaboration Projects (IAF-ICP) Funding Initiative, as well as cash and in-kind contribution from the industry partner, HP Inc., through the HP-NTU Digital Manufacturing Corporate Lab (I1801E0028). This work is also partially supported by the Ministry of Education, Singapore, under its Academic Research Fund Tier 2 (MOE2019-T2-1-071), and Nanyang Technological University, Singapore, under its NAP (M4082282).

978-3-9819263-7-8/DATE23/© 2023 EDAA

TABLE I: Comparison with SOTA pruning approaches on ImageNet. The baseline network is ResNet50. {d, w, r} indicate the pruned dimensions in different methods.

Method	d	w	r	MACs (B)	Acc@1 (%)	Acc@5 (%)
ResNet50 [7]				4.10	76.80	93.38
DECORE-4 [8]		✓		1.19	69.71	89.37
GAL-1 [9]	✓	✓		1.58	69.82	89.75
Bilinear [10]			✓	1.10	69.97	89.19
HAP [11]		✓		1.34	71.18	-
Taylor [6]		✓		1.34	71.69	-
HRank [3]		✓		1.55	71.98	91.09
DBP-0.5 [4]	✓			2.05	72.44	-
EMNAPE-S (ours)	✓	✓	✓	**1.05**	**73.07**	**91.18**
GAL-0.5 [9]	✓	✓		2.33	71.95	90.94
Bilinear [10]			✓	2.53	73.40	91.30
RANet [12]			✓	2.30	74.00	-
Taylor [6]		✓		2.25	74.50	-
DBP-0.4 [4]	✓			2.56	74.74	-
HRank [3]		✓		2.30	74.98	92.33
DR-ResNet50 [5]			✓	2.30	75.30	92.20
EMNAPE-M (ours)	✓	✓	✓	**2.24**	**75.68**	**92.79**
SSS-32 [13]	✓	✓		2.82	74.18	91.91
Bilinear [10]			✓	3.00	74.30	91.90
HAP [11]		✓		2.71	75.12	-
Taylor [6]		✓		**2.66**	75.48	-
PFP-A [14]		✓		3.70	75.90	92.80
DECORE-8 [8]		✓		3.54	76.31	93.02
EMNAPE-L (ours)	✓	✓	✓	2.87	**76.34**	**93.20**

on-device inference latency. To address this problem, in the subsequent stage, we design a novel importance metric to further compare units of different dimensions according to their impacts on model computation, accuracy, and inference latency. The proposed importance metric for inter-dimensional evaluation is formulated as follows:

$$I_{out} = \frac{\Delta A(u)}{\alpha \Delta M(u) + (1-\alpha)\Delta T(u)} \tag{2}$$

where I_{out} represents the inter-dimensional importance score of u. ΔA, ΔM, and ΔT denote the changes in accuracy, model computation, and inference latency caused by pruning u, respectively. $\alpha \in [0, 1]$ is a hyperparameter to control the contribution of ΔM and ΔT, which is empirically set to 0.5 in this paper. Although collecting ΔA, ΔM, and ΔT requires considerable training and deployment costs, we only perform the inter-dimensional evaluation on the most redundant unit of each dimension, and thus the cost of inter-dimensional evaluation is mitigated significantly. With the two-stage evaluation pipeline, we can efficiently and comprehensively identify the redundancy in the three dimensions. Finally, we present a heuristic pruning algorithm to progressively prune the three dimensions of CNNs. In each pruning iteration, we will utilize the proposed two-stage evaluation strategy to identify the most redundant unit from the three dimensions, and then the unit will be safely pruned. At the end of each pruning iteration, we will fine-tune the model for 1 epoch to more accurately estimate the importance of each unit for the next pruning iteration.

III. Experiments

We evaluate our approach on ImageNet and summarize the results in TABLE I, which demonstrates that our approach outperforms other competitors across a wide spectrum of model computation. Specifically, in the low compute regime,

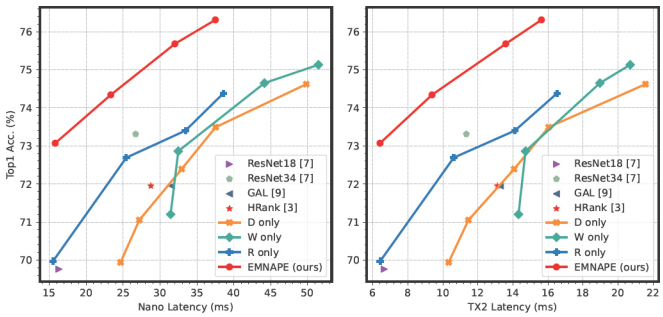

Fig. 2: Comparison of inference latency on Nano and TX2.

EMNAPE-S observes 3.36% higher top-1 accuracy with about 12% fewer MACs than DECORE-4 [8]. In the highest MACs regime, EMNAPE-L surpasses SSS-32 [13] with 2.16% higher top-1 accuracy. In addition, we also evaluate the run-time latency of models obtained from different pruning frameworks on two popular edge platforms: 1) Jetson Nano and 2) Jetson TX2. The experimental results are shown in Fig. 2, which reveals that our approach achieves the best on-device efficiency on both platforms. Specifically, our method achieves 3.7% higher accuracy than HRank [3] with similar latency on Jetson TX2. On Jetson Nano, our method also observes 1.12% higher accuracy than GAL [9] with only 50% latency budget.

IV. Conclusion

In this paper, we present a multi-dimensional pruning framework, EMNAPE, to prune CNNs for higher inference efficiency on embedded devices. With our two-stage evaluation strategy, we effectively identify the computational redundancy in the three dimensions. Subsequently, our heuristic pruning algorithm efficiently compress CNN models towards better trade-offs between accuracy and efficiency. Extensive experiments validate the superiority of EMNAPE over existing pruning approaches.

References

[1] J. Deng *et al.*, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009.

[2] T.-Y. Lin *et al.*, "Microsoft coco: Common objects in context," in *ECCV*, 2014.

[3] M. Lin *et al.*, "Hrank: Filter pruning using high-rank feature map," in *CVPR*, 2020.

[4] W. Wang *et al.*, "Dbp: Discrimination based block-level pruning for deep model acceleration," *arXiv preprint arXiv:1912.10178*, 2019.

[5] M. Zhu *et al.*, "Dynamic resolution network," in *NeurIPS*, 2021.

[6] P. Molchanov *et al.*, "Importance estimation for neural network pruning," in *CVPR*, 2019.

[7] K. He *et al.*, "Deep residual learning for image recognition," in *CVPR*, 2016.

[8] M. Alwani *et al.*, "Decore: Deep compression with reinforcement learning," in *CVPR*, 2022.

[9] S. Lin *et al.*, "Towards optimal structured cnn pruning via generative adversarial learning," in *CVPR*, 2019.

[10] M. Sandler *et al.*, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *CVPR*, 2018.

[11] S. Yu *et al.*, "Hessian-aware pruning and optimal neural implant," in *WACV*, 2022.

[12] L. Yang *et al.*, "Resolution adaptive networks for efficient inference," in *CVPR*, 2020.

[13] Z. Huang and N. Wang, "Data-driven sparse structure selection for deep neural networks," in *ECCV*, 2018.

[14] L. Liebenwein *et al.*, "Provable filter pruning for efficient neural networks," in *ICLR*, 2020.