# Achieving an optimal group structure in a neural architecture search

W. Seo,[1] J. Park,[1] W. Lim,[1] D. Kim,[1] and J. Lee[2,✉]

[1]*School of Computer Science and Engineering, Chung-Ang University, Heukseok-Ro, Dongjak-gu, Seoul, Republic of Korea*

[2]*Department of Artificial Intelligence, Chung-Ang University, Heukseok-Ro, Dongjak-gu, Seoul, Republic of Korea*

✉Email: curseor@cau.ac.kr

The method proposed in this letter searches for an effective group structure of group convolutions in a convolutional neural network that can improve the classification accuracy. The model's group structure is obtained using an effective differential neural architecture search. Our code can be accessed at https://github.com/minercode625/grunas.git.

*Introduction:* Neural architecture search (NAS) is a methodology that can automate the design process of convolutional neural networks (CNNs), thus minimising human intervention [1]. The method can discover model architectures that can achieve high accuracy on a given dataset [2]. Compared with existing approaches, differential neural architecture search (DNAS) achieves superior effectiveness owing to a gradient-based optimisation that simultaneously optimises the model architecture and corresponding parameters [3]. Particularly, most DNAS-based methods concentrate on searching for effective operators among other possible operators. However, critical issues still remain unsolved, such as determining the group size in group convolution [4], which is dependent on the decisions of experts. Although each task or dataset can possess a different optimal size in each group convolution, ignoring the efficacy of the optimal size limits the model learning accuracy. To the best of our knowledge, this is the first study that attempts to identify the optimal group sizes for all group convolutions by DNAS. We confirm that the model obtained from the proposed method outperforms conventional models in terms of the accuracy, latency, and model size.

*Proposed method:* DNAS executes a search process using a gradient-based optimiser by relaxing the categorical choice in a model structure into a continuous structure. It simultaneously trains the model parameters and evaluates the importance weights of all possible operators in the model by using backpropagated gradients. The gradient-based optimiser increases the weights of the effective operators, and towards the end of the search process, the operators with the largest weights are selected to form the model architecture.

The DNAS framework formulates the NAS problem as follows:

$$\min_{\alpha\in\mathcal{A}} \min_{w_a} \mathcal{L}(\alpha, w_\alpha), \tag{1}$$

where $\mathcal{L}$, $\mathcal{A}$ and $w_\alpha$ denote the DNAS loss function, architecture space and model parameters, respectively, under current model architecture $\alpha$. Let an ordered set $H = \{h_1, h_2, \ldots, h_n\}$ be a group structure consisting of the sizes of all group convolutions, where $h_k$ denotes the group size in the $k$th group convolution. We modified (1) to search for an optimal $H$ as follows:

$$\min_{H\in\mathcal{H}} \min_{w_H} \mathcal{L}(H, w_H), \tag{2}$$

where $w_H$ denotes model parameters when the group structure is $H$ and $\mathcal{H}$ is a group structure space comprising all possible $H$. Particularly, we reformulate the model loss function as follows to avoid excessively increasing the number of parameters according to the searched group structure:

$$\mathcal{L}(H, w_H) = CE(H, w_H) \cdot P(H)^\beta, \tag{3}$$

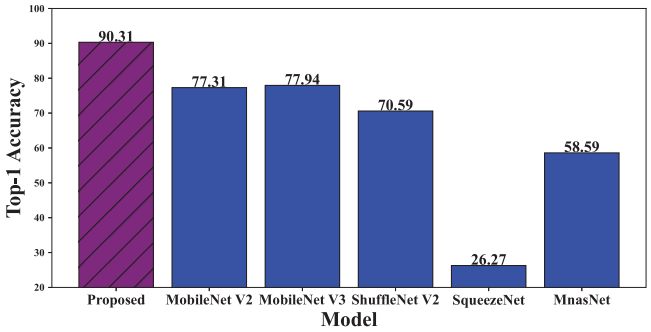


**Fig. 1** *Top-1 accuracy score of the proposed method and conventional models computed using the CIFAR-10 dataset, where the obtained group structure is* $H = \{4, 1, 1, 2, 2, 3, 2, 4, 4, 6\}$

where $CE$ denotes the model cross-entropy loss, and the exponent coefficient $\beta$ adjusts the magnitude of the latency term. $P$ denotes the number of parameters when the group structure is the $H$, which is given by

$$P(H) = \sum_{h_i\in H} m(h_i), \tag{4}$$

where $m(h_i)$ denotes the number of parameters when the group size is $h_i$. A set $G = \{g_1, g_2, \ldots\}$ is given such that each $g_k$ represents a candidate of group numbers and $h_i \in G$. The proposed method optimises a backbone model using the loss function based on the weighted sum of all candidate group convolutions. Thus, (4) can be written as

$$P(H) = \sum_{k=1}^{|H|} \sum_{g\in G} GS(\theta_{k,g}) m(g), \tag{5}$$

where $\theta_{k,g}$ represents the importance weight of the $k$th group convolution when the size is $g$. In addition, the functions $GS$ and $m$, respectively, represent a Gumbel Softmax function and number of required group convolution parameters, where the group size is $g$ [2]. As (5) is a discrete function, the discrete variable $\theta$ is transformed into a continuous random variable using the Gumbel Softmax function. Finally, the proposed method can be formally defined as a weighted sum as follows:

$$\min_{\theta} \min_{w_H} \mathsf{E}_{H\sim P_\theta}\{CE(H, w_H) \cdot P(H)^\beta\}. \tag{6}$$

After the search process and optimisation of all the group convolution weights, the proposed method determines that the group convolution with the highest weight is the optimal operator.

*Experimental results:* To demonstrate the effectiveness of proposed method, we conducted experiments compared with four well-known models, MobileNet V2 [5], MobileNet V3 [6], ShuffleNet V2 [7], SqueezeNet [8], and MnasNet [1], searched using the NAS originating from MobileNet. We conducted the comparison experiments on the CIFAR-10 image dataset [9], where all models were trained using 50,000 images and tested using 10,000 images. We set the network input resolution and $\beta$ to 36-by-36 and 0.8, respectively, and trained each model from scratch for 100 epochs. $G$ was set to $\{1, 2, 3, 4, 6\}$ [7], and each $\theta$ per group convolution was uniformly set to 0.20, initially reflecting $|G|$. In addition, we used a backbone model composed of 10 ShuffleNet units [4] which contain group convolutions inside, but note that the proposed method can be generalised to use any backbone models that contains a group convolution. Two hyper-parameters in the Gumbel Softmax function, namely initial temperature and annealing rate, were set to 5.0 and $e^{-0.645}$, respectively [2].

Figure 1 depicts the accuracies of the five conventional models and that of our model obtained on the CIFAR-10 test dataset. As the backbone model possesses 10 ShuffleNet units, the proposed method searched for the optimal $H = \{h_1, h_2, \ldots, h_{10}\}$, $h_i \in G$. Consequently, it concluded that the model was most effective when the group structure was $H = \{4, 1, 1, 2, 2, 3, 2, 4, 4, 6\}$. The proposed model achieved the highest top-1 accuracy of 90.31%, whereas the SqueezeNet and

*Table 1. CIFAR-10 performances of the proposed method compared with that of conventional models in terms of the latency, number (#) of parameters, model size, and top-1 accuracy (Accuracy)*

| Model | GPU latency | # of parameters | Model size | Accuracy |
|---|---|---|---|---|
| MobileNet V2 | 20 ms | 2.24 M | 8.53 MB | 77.31 |
| MobileNet V3 | 23 ms | 4.21 M | 16.08 MB | 77.94 |
| ShuffleNet V2 | 19 ms | 0.35 M | 1.34 MB | 70.59 |
| SqueezeNet | 7 ms | 0.73 M | 2.78 MB | 26.27 |
| MnasNet | 14 ms | 3.12 M | 11.88 MB | 58.59 |
| **Proposed** | 4 **ms** | 0.16 **M** | 0.64 **MB** | 90.31 |

MnasNet achieved 26.27% and 58.59%, respectively, and the accuracies of the other models were lower than 80% as well. The conventional models demonstrated relatively poorer performances than their originally reported results because they were not pre-trained but trained from scratch. In contrast, the proposed model achieved faster convergence in the same situation than the other models. Table 1 lists the comparison results in terms of the GPU latency, number of parameters, model size, and top-1 accuracy. As the proposed loss function avoids excessively increasing the total number of parameters, the proposed model can achieve lower GPU latency as well as a smaller number of parameters and model size than those produced by conventional models. For example, it was five times faster than MobileNet V2 in terms of the GPU latency and 25.1 times smaller than MobileNet V3 in terms of the model size. Specifically, the number of group convolution parameters is proportional to the number of input/output channels and inversely proportional to the group size. The model group size proposed at the bottom of the model ($\{h_8, h_9, h_{10}\} = \{4, 4, 6\}$), wherein the number of input/output channels that was considerably increased previously, was reduced by (6).

*Conclusion and future work:* We proposed a novel NAS method that searches for an effective group structure in group convolutions. The model developed by the proposed method was demonstrated to be effective in that it outperformed conventional models. In a future study, we plan to apply the proposed method to large deep learning models and investigate subsequent changes in those models.

*Acknowledgements:* This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (2021-0-01341, Artificial Intelligence Graduate School Program (Chung-Ang University)) and the National Research Foundation of Korea (NRF) grant funded by the Korea government(MSIT) (No. 2020R1A2C101357511).

*Conflict of interest:* The authors have declared no conflict of interest.

*Data availability statement:* The data that support the findings of this study are openly available in at http://www.cs.toronto.edu/~kriz/cifar.html


Received: *11 July 2022* Accepted: *11 July 2022*
doi: 10.1049/ell2.12585

**References**

1 Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A., Le, Q.V.: Mnasnet: Platform-aware neural architecture search for mobile. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2820–2828. IEEE, Piscataway, NJ (2019)
2 Wu, B., Dai, X., Zhang, P., Wang, Y., Sun, F., Wu, Y., Tian, Y., Vajda, P., Jia, Y., Keutzer, K.: Fbnet: Hardware-aware efficient convnet design via differentiable neural architecture search. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 10 734–10 742. IEEE, Piscataway, NJ (2019)
3 Liu, H., Simonyan, K., Yang, Y.: Darts: Differentiable architecture search. Preprint, arXiv:1806.09055 (2018)
4 Zhang, X., Zhou, X., Lin, M., Sun, J.: Shufflenet: An extremely efficient convolutional neural network for mobile devices. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 6848–6856. IEEE, Piscataway, NJ (2018)
5 Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.-C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 4510–4520. IEEE, Piscataway, NJ (2018)
6 Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., Le, Q.V., Adam, H.: Searching for mobilenetv3. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1314–1324. IEEE, Piscataway, NJ (2019)
7 Ma, N., Zhang, X., Zheng, H.-T., Sun, J.: Shufflenet v2: Practical guidelines for efficient CNN architecture design. In: Proceedings of the European Conference on Computer Vision, pp. 116–131. (2018)
8 Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J., Keutzer, K.: SqueezeNet: Alexnet-level accuracy with 50x fewer parameters and < 0.5 MB model size. Preprint, arXiv:1602.07360 (2016)
9 Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Tech. Rep., University of Toronto, Toronto, ON (2009)