Ensemble Knowledge Guided Sub-network Search and Fine-tuning for Filter Pruning

Seunghyun Lee[0000−0001−7139−1764] and Byung Cheol Song[0000−0001−8742−3433]

Inha University, Incheon, Republic of Korea
lsh910703@gmail.com, bcsong@inha.ac.kr

Abstract. Conventional NAS-based pruning algorithms aim to find the sub-network with the best validation performance. However, validation performance does not successfully represent test performance, i.e., potential performance. Also, although fine-tuning the pruned network to restore the performance drop is an inevitable process, few studies have handled this issue. This paper provides a novel Ensemble Knowledge Guidance (EKG) to solve both problems at once. First, we experimentally prove that the fluctuation of loss landscape can be an effective metric to evaluate the potential performance. In order to search a sub-network with the smoothest loss landscape at a low cost, we employ EKG as a search reward. EKG utilized for the following search iteration is composed of the ensemble knowledge of interim sub-networks, i.e., the by-products of the sub-network evaluation. Next, we reuse EKG to provide a gentle and informative guidance to the pruned network while fine-tuning the pruned network. Since EKG is implemented as a memory bank in both phases, it requires a negligible cost. For example, in the case of ResNet-50, just 315 GPU hours are required to remove around 45.04% of FLOPS without any performance degradation, which can operate even on a low-spec workstation. the source code is available at here.

1 Introduction

Network pruning is attracting a lot of attention as a lightweight technique to reduce computation and memory cost by directly removing parameters of deep neural networks (DNNs). In particular, filter pruning is advantageous in accelerating using the basic linear algebra subprograms (BLAS) library because it eliminates parameters in units of filters. Recently, filter pruning has been regarded as a kind of neural architecture search (NAS), that is, the sub-network search process. Actually, some methods using the existing NAS algorithms succeeded in finding a high-performance pruned network [6,25,24,40]. Based on 'supernet' capable of dynamic inference, the other methods significantly reduced computational complexity while maintaining high performance of NAS-based pruning algorithm [2,45,11,36].

Acknowledgments: This work was supported by IITP grants (No.2021-0-02068, AI Innovation Hub and RS-2022-00155915, Artificial Intelligence Convergence Research Center(Inha University)), and the NRF grants (No. 2022R1A2C2010095 and No. 2022R1A4A1033549) funded by the Korea government (MSIT).

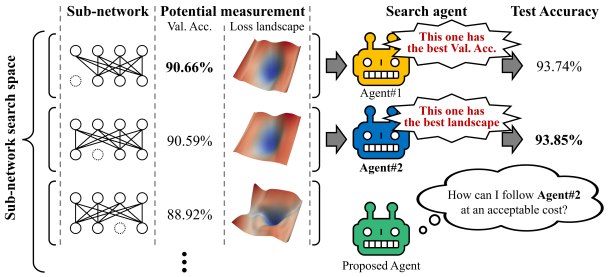

Fig. 1. Conceptual visualization showing the performance change of the search algorithm according to reward and the intrinsic goal of the proposed agent

In general, NAS-based pruning algorithms consist of a search phase and a fine-tuning phase. We paid attention to the fact that prior arts seldom dealt with critical factors in each phase. First, note that rewards used in the search phase so far are not accurate enough to find the optimal sub-network. NAS-based pruning usually samples the validation set from the training dataset and uses the validation performance as a reward. In other words, it is assumed that validation performance has a high correlation with test performance, i.e., potential performance. Although this approach improves the search speed greatly, it implies the possibility of overfitting the sub-network to the validation set. Despite such a risk, most previous works adopted validation performance as the reward without any doubt. Second, there were no prior arts considering the characteristics of the pruned network in the fine-tuning phase. During the filter pruning procedure, some information loss is unavoidable. To recover the information loss, we should fine-tune the pruned network. Since the pruned network still possesses the information of the pre-trained network, an approach differentiated from randomly-initialized networks is required. However, prior arts employed general learning strategies used when learning pre-trained networks as they are or adopted primitive knowledge distillation (KD).

To solve two problems mentioned above, this paper presents a single solution, i.e., Ensemble Knowledge Guidance (EKG). In general, a student network constrained by teacher knowledge has a flatter local minima [27,38,3] and faster convergence speed [39], compared to ordinary optimization algorithms. Based on these characteristics, we formulate a way to improve the search and fine-tuning phases by using EKG as follows: First, we show that loss landscape fluctuation is a useful tool to evaluate the potential performance of sub-network. Fig. 1 visualizes the concept. The sub-network with the highest potential performance is expected to be sufficiently generalized as well as to have a fast learning speed. Since the smoothness of loss landscape can estimate both factors, i.e., generalization and learning speed [20,34,9], we employ it to evaluate sub-networks. However, since loss landscape-based reward requires massive computational cost, it is impractical to utilize the smoothness of loss landscape as a search reward. So, we use EKG to find an optimal sub-network with the smoothest loss landscape at an acceptable cost. Based on the property of KD, EKG implicitly selects a sub-network with a smoother loss landscape and discovers the optimal sub-network more precisely than previous works that solely use the validation loss. Here, as

the source of knowledge, the output features of interim sub-networks, i.e., by-products of each search iteration, are stored and ensembled in the memory bank. Therefore, EKG incurs only a negligible cost because we don't have to infer the numerous genuine teacher networks in every training iteration.

Furthermore, EKG is applied once again to fine-tune the pruned network. As mentioned above, KD is an effective way to improve a small network. However, as the pruning rate increases, the gap between the pre-trained and pruned networks also increases, which makes it difficult to transfer knowledge [27]. To bridge the performance gap, we adopt interim sub-networks once again as teacher-assistant networks and build a memory bank with their knowledge. Then, according to the learning status of the pruned network, the knowledge of the memory bank is ensembled and transferred. Since the knowledge always maintains an appropriate gap with the pruned network, over-constraints can be avoided.

As a result, the proposed EKG can improve the performance of the pruned network while affecting both phases with a marginal cost increase. For example, when pruning ResNet-50, 45.04% FLOPS can be removed without any performance drop in only 315 GPU hours, which is an impressive result considering that our experiments are performed on a low spec workstation. Finally, the contributions of this paper are summarized as follows:

- As a new tool to measure the potential performance of sub-network in NAS-based pruning, the smoothness of loss landscape is presented. Also, the experimental evidence that the loss landscape fluctuation has a higher correlation with the test performance than the validation performance is provided.
- EKG is proposed to find a high potential sub-network and boost the fine-tuning process without complexity increase, which makes a high-performance light-weighted neural network democratized.
- To our knowledge, this paper provides the world-first approach to store the information of the search phase in a memory bank and to reuse it in the fine-tuning phase of the pruned network. The proposed memory bank contributes to greatly improving the performance of the pruned network.

2 Related Work

This section reviews the research trend of filter pruning. Existing filter pruning algorithms can be categorized into two groups. The first approach is filter importance scoring (FIS). FIS scores the importance of each filter according to a specific criterion and removes lower-score filters. Early FIS techniques adopted filter norms as criteria [13]. However, they faced with a problem of 'smaller-norm-less-importance criteria.' So, various scoring techniques were proposed to solve this problem, e.g., gradient-norm [28,44], geometric relation [16,40], and graph structure [19,41]. Although the latest FIS techniques improved the performance of the pruned network, a question still remains as to whether the existing filter scores are closely related to the performance degradation.

The second is a NAS-based approach and is more intuitive than the first. NAS-based methods define the number of filters as a search space and search for

the optimal sub-network by learning and evaluating each sub-network [6,25,24]. Here, the performance itself is considered a score. However, the NAS-based methodology inevitably requires a huge cost because each sub-network must be actually trained. Thus, supernet was proposed to reduce this massive cost. A supernet in which a low index filter has higher importance than a high index filter is a network trained to enable dynamic inference [47,11,45]. Once a supernet is trained, the search agent can explore the search space by free, so the search cost can be greatly reduced. Thus, supernet-based pruning has become the most dominant methodology. For example, [36,37] achieved state-of-the-art (SOTA) performance by adopting a more sophisticated search algorithm, and [46,45] presented a few effective methods for training supernets. However, since a supernet is not publicly available in general, it must be re-trained every time. In other words, supernet-based methods have a fatal drawback in that they cannot use well-trained networks [10,43].

The most important virtue of the pruning algorithm is to find a sub-network that minimizes information loss, but the process of restoring it is just as important. Surprisingly, how to fine-tune the pruned network has been rarely studied. For example, most pruning algorithms adopted a naive way of training the pruned network according to a general training strategy. If following such a naive way, a huge cost is required for fine-tuning. As another example, a method to re-initialize the pruned network according to lottery-ticket-hypothesis was proposed [7]. This method can guide the pruned network fallen into the local minima back to the global minima, but its learning cost is still huge. The last example is using KD [17]. In general, the pre-trained network has a similar structure to the pruned network, but has a higher complexity and is more informative. So, we can intuitively configure the pre-trained and pruned networks as a teacher-student pair. Recently, a few methods have been proposed to improve the performance of the pruned network with the pre-trained network as a teacher [6]. However, if the pruning rate increases, that is, if the performance gap between teacher and student increases, the information of the teacher network may not be transferred well [27]. Also, since the pruned network has information about the target dataset to some extent, unlike general student networks, fine-tuning for the pruning algorithm is required.

3 Method: Ensemble knowledge guidance

3.1 Accurate potential performance evaluation

Conventional NAS-based pruning iteratively eliminates redundant filter set θ from the pre-trained network's filter set Θ_0 until the target floating point operations (FLOPs) are reached. This process is expressed by

$$\Theta_{i+1} = \Theta_i \setminus \theta_i^* \tag{1}$$

$$\theta_i^* = \underset{\theta_i}{\operatorname{argmax}}\ \mathcal{R}\left(\Theta_i, \theta_i\right) \tag{2}$$

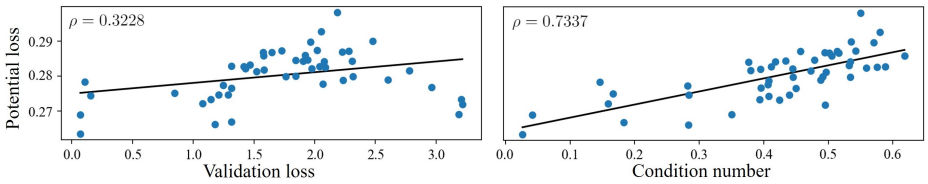

Fig. 2. (Left) Potential loss vs. validation loss (Right) Potential loss vs. condition number. 50 sub-networks of ResNet-56 trained on CIFAR10 were used for this experiment

where i is the search step, $\setminus$ is the set subtraction, and $\mathcal{R}$ indicates the reward of the search agent. It is costly to handle all filter combinations, so the agent usually divides the search space into intra-layer and inter-layer spaces. Then, the redundant filter set candidate $\phi_{i,l}$ is determined in each layer, and the candidate with the highest reward is selected. If a supernet is used as a pre-trained network, a candidate of each layer can be determined at no cost. As a result, the search process is regarded as a task that evaluates candidates, and is re-written by

$$\Theta_{i+1} = \Theta_i \setminus \phi_{i,l^*} \tag{3}$$

$$\begin{aligned} l^* &= \underset{l}{\operatorname{argmax}}\ \mathcal{R}(\Theta_i, \phi_{i,l}) \\ \text{s.t. } \Phi_i &= \{\phi_{i,l} | 1 \leq l \leq L\} \end{aligned} \tag{4}$$

where L stands for the number of layers in DNN.

Among prior arts, some concentrated on how to construct candidates while maintaining the above-mentioned framework [47,46,45,2], and others presented new search agents to select the best candidate [37,36]. On the other hand, most of the previous studies employed sub-network's performance as a search reward $\mathcal{R}$ without any doubt. Since it is too heavy to use the entire training dataset for validation, the validation set $\mathcal{D}^{val}$ is sampled from the training dataset. Therefore, assuming that the loss function $\mathcal{L}$ is used as a metric to measure performance and $\phi_{i,l}$ is removed from Θ_i by a certain NAS-based pruning algorithm, the search reward is defined by

$$\mathcal{R}(\Theta_i, \phi_{i,l}) = -\mathcal{L}(\mathcal{D}^{val}; \Theta_i \setminus \phi_{i,l}) \tag{5}$$

The reason to search for the sub-network with the highest validation performance is that its potential performance after fine-tuning is expected to be high. In other words, it is assumed that validation performance is highly correlated with potential performance. To examine whether this assumption holds, we randomly sampled sub-networks of ResNet-56 [14] trained on CIFAR10 [18] and measured the validation performance and test performance after fine-tuning. Specifically, Pearson Correlation Coefficient [35] (PCC) $\rho_{X,Y}$ of the validation loss (X) and potential loss (Y) was calculated. For the detailed training configuration of this experiment, please refer to the supplementary material. The left plot of Fig. 2 shows that the correlation between the two losses is not so high.

Especially, note that the validation loss has high variance in the low potential loss region of the most interest. This phenomenon makes it difficult to adopt the validation loss as a reliable reward. Since the search process is a sort of optimization, exploring the validation set makes the sub-network fit the validation set. In other words, there is a risk of overfitting. Therefore, instead of simply measuring validation performance, we need another indicator to measure generality.

We introduce loss landscape [20] as a means to analyze the generality of sub-networks. The more the loss landscape is smooth and close to convex, the more the network is robust and easy to be trained. Therefore, quantitatively measuring fluctuations in the loss landscape can determine how well the network is generalized, which provides higher-order information than performance. Because a network with fewer filters has relatively large fluctuations in the loss landscape, the loss landscape of the sub-network becomes more complex as pruning progresses. However, if the information of a certain filter is redundant, the generality error as well as fluctuations in the loss landscape do not increase. Based on this insight, we assume that a sub-network can be evaluated through the fluctuation of the loss landscape. To verify whether this assumption is valid, we examine the PCC of loss landscape fluctuation (X) and potential loss (Y). As an index representing the loss landscape fluctuation, we employed the condition number (CN), i.e., the ratio of minimum and maximum eigenvalues of the Hessian [31,20], which is defined by

$$\text{Condition number} = \left|\lambda^{\min}/\lambda^{\max}\right| \tag{6}$$

The right plot of Fig. 2 shows that CN has a higher correlation with potential loss than validation loss. In particular, note that the variance of CN is very small in the low potential loss region. This proves that CN is a more reliable indicator to evaluate sub-networks. Based on this experimental result, the next subsection designs a search process to select a sub-network with a smoother loss landscape.

3.2 Search with ensemble knowledge guidance

CN must be a better indicator to evaluate potential performance than validation performance. However, since the Hessian matrix required to calculate CN causes a huge computation cost, it is burdensome to directly use CN as a search reward.

To design a new sub-network search process of a reasonable cost, we adopt EKG. KD is widely used to enhance the performance of a small-size network (student) with a large-size network (teacher). Many researchers have pointed out that the performance of a student is improved because the loss landscape of the student gets smoother by receiving the teacher's knowledge, that is, the generality error of the student is reduced [27,38,3]. Student networks are saturated with a smoother loss landscape under multi-directional constraints due to knowledge and target loss. However, if the pruning rate gets higher, a performance gap between the pre-trained network and the sub-network gets larger, making teacher knowledge not transferred effectively [27]. Fortunately, interim sub-networks have suitable properties as teacher-assistant networks [27]. In detail,

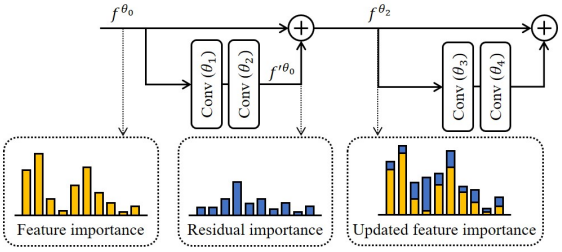

Fig. 3. An example of a position sensing a feature map f^{θ_2} for scoring a filter θ_2 in the proposed method. Compared with f'^{θ_2}, which only grasps residual importance, f^{θ_2} can observe that importance is well measured regardless of architecture characteristics

interim sub-networks have intermediate performance between the pre-trained and pruned networks and are computationally efficient because there is no additional training cost. So, we build a memory bank and ensemble the knowledge of interim sub-networks at every step to keep the knowledge guidance at the middle point of the pre-trained network and the current sub-network. Finally, $\mathcal{R}$ is re-defined by

$$\mathcal{R}(\Theta_i, \phi_{i,l}) = -\mathcal{L}(\mathcal{D}^{val}; \Theta_i \setminus \phi_{i,l}) - \mathcal{L}(\mathcal{T}_i; \Theta_i \setminus \phi_{i,l}) \tag{7}$$

$$\mathcal{T}_i = \frac{1}{i+1} \sum_{j=0}^{i} \mathcal{O}(\mathcal{D}^{val}; \Theta_j) \tag{8}$$

where $\mathcal{O}(\cdot;\cdot)$ indicates the inference result.

Based on the proposed reward, we reconfigure the existing search process as follows. First, a score is given to each filter according to Eq. (7), and filters of low scores are selected as candidates. Here, in order to more accurately evaluate the importance of θ, the input feature map of the next layer f^{θ} is used as in Fig. 3. Since f^{θ} has both the information generated from θ and the characteristics of the network architecture, it can more accurately represent the importance of each filter. Using the well-known Taylor expansion [28,44], we propose the following scoring function:

$$\mathcal{S}(\theta) = \left| \frac{\partial \mathcal{R}(\Theta_i, \cdot)}{\partial f^{\theta}} f^{\theta} \right| \tag{9}$$

The detailed derivation process of the above formula is depicted in the supplementary material. Next, filters of lower scores as much as a specific ratio r for each layer become candidates, and a candidate with the highest reward is selected through a greedy search. Finally, the optimal sub-network Θ^* is searched iteratively.

In fact, the concept of transferring knowledge of a large sub-network to a small sub-network is already widely used for learning supernets [47,45,11,2,36]. We can say that the proposed reward based on EKG inherits this concept. Here,

Algorithm 1: The proposed pruning algorithm

Input : $\Theta_0, \mathcal{D}^{train}, r$ **Output** : Fine-tuned network
Sample $\mathcal{D}^{subset}$ and $\mathcal{D}^{val}$ in $\mathcal{D}^{train}$ and fine-tune Θ_0 in an epoch on $\mathcal{D}^{subset}$
Store initial ensemble knowledge $\mathcal{T}_0$
Repeat
    Compute filter importance scores $\mathcal{S}(\theta)$ by Eq. (9)
    Sample candidates Φ_i in each layers with r.
    Select $\phi^*_{i,l}$ by Eq. (4) that maximizes Eq. (7).
    Get next pruned network Θ_{i+1} by Eq. (3).
    Update ensemble knowledge $\mathcal{T}_{i+1}$ by Eq. (8)
    $i = i + 1$
Until FLOPs reduction rate reaches the goal
$\Theta^* = \Theta_i$
Build memory bank by Eq. (10)
Repeat
    Get training sample in $\mathcal{D}^{train}$ and apply two augmentation functions.
    Minimize loss function in Eq. (12).
Until Training is done

scoring based on Taylor expansion can compute the scores of all filters at once, so it not only requires a much lower cost than learning the supernet itself but also has the advantage of using well-trained networks as they are.

3.3 Fine-tuning with ensemble knowledge guidance

Even though a sub-network with high potential is available, the performance of the sub-network can be maximized through fine-tuning. So, we once again utilize EKG. As mentioned in Sec. 3.2, the pruning process generates many qualified teacher networks, i.e., interim sub-networks. If this knowledge is transferred to the pruned network again, the effect of inverse-tracking the information loss of the pruning process can be accomplished. However, acquiring knowledge by inferencing of all teacher networks at every iteration requires a huge computational cost. To transfer teacher knowledge efficiently, we propose to sample interim sub-networks with a uniform performance gap and store their knowledge in a memory bank. This entire procedure is expressed by

$$\mathcal{M} = \{\mathbf{M}_k = \mathcal{O}(\mathcal{D}^{train}, \mathcal{T}_k) | 1 \leq k \leq K\} \tag{10}$$

$$\mathcal{T}_k = \underset{\Theta_i}{\operatorname{argmin}} \left| \frac{K-k}{K}\mathcal{L}(\Theta^*) + \frac{k}{K}\mathcal{L}(\Theta_0) - \mathcal{L}(\Theta_i) \right| \tag{11}$$

where K indicates the number of interim sub-networks to be sampled.

Memory bank knowledge smoothly bridges the performance gap between pre-trained and pruned networks, but fixed knowledge can often cause overfitting. To resolve this side-effect, we employ contrastive learning. Contrastive learning is to minimize the gap between two data representations to which different

augmentations are applied [4,10]. So, it allows DNNs to learn more generalized representations. We set memory bank knowledge as the center of the augmented data distribution and transfer it into the sub-network. As in the search process, we select the teacher network from $\mathcal{M}$ according to the performance of the pruned network, and then the ensemble knowledge is transferred. Therefore, the loss function $\mathcal{L}^{ft}$ for fine-tuning is defined by

$$\mathcal{L}^{ft} = \sum_{a=1}^{2} \mathcal{L}(\mathcal{D}^{train}, \mathcal{A}_a; \Theta^*) + \mathcal{L}(\overline{\mathbf{M}}; \Theta_i \setminus \phi_{i,l}) \tag{12}$$

$$\overline{\mathbf{M}} = \mathbb{E}\left\{ \mathbf{M}_k \,\middle|\, \begin{array}{l} 1 \leq k \leq K \\ \mathcal{L}(\mathbf{M}_k) \leq \mathcal{L}(\mathcal{D}^{train}; \Theta^*) \end{array} \right\} \tag{13}$$

where $\mathcal{A}$ stands for augmentation function. The proposed fine-tuning strategy injects extra information into the pruned network at almost no cost. Also, it not only improves the learning speed but also has a regularization effect so that the pruned network has higher performance. Algorithm 1 summarizes the proposed method. The proposed search process has the following differentiation points from the existing supernet-based search algorithm.

• Because of the ease of using the pre-trained network and the accurate potential performance evaluation, a better sub-network can be obtained.
• The potential of sub-network is maximized by transferring the information lost in the search phase back to the fine-tuning phase, resulting in high performance.
• Since the proposed search process has almost no additional cost compared to prior arts, it can operate even on a low-performance computer system.

4 Experimental results

This section experimentally proves the performance of EKG. First, we verify that CN can be a metric that evaluates sub-networks more accurately than validation performance. Second, an ablation study is given. Third, we prove that the proposed method accomplishes more effective pruning than conventional methods in various configurations and achieves SOTA performance at a reasonable cost. The datasets for experiments were CIFAR10, CIFAR100 [18], and ImageNet [32]. ResNet family [14] and MobileNet-v2 [33] were used as network architectures. The proposed method was implemented in Tensorflow [1]. The CIFAR pre-trained network was trained by ourselves, and the ImageNet pre-trained network is a version released in Tensorpack [42]. Detailed hyper-parameters for training are shown in the supplementary material. GPU hours were calculated on GTX1080Ti, and 1 to 2 GPUs were utilized, depending on the architecture.

4.1 Empirical analysis for loss landscape-based evaluation

This subsection proves that the proposed method effectively searches the sub-network of smoother loss landscape. Unfortunately, direct comparison with existing supernet-based search algorithms is unreasonable. Instead, we assigned a

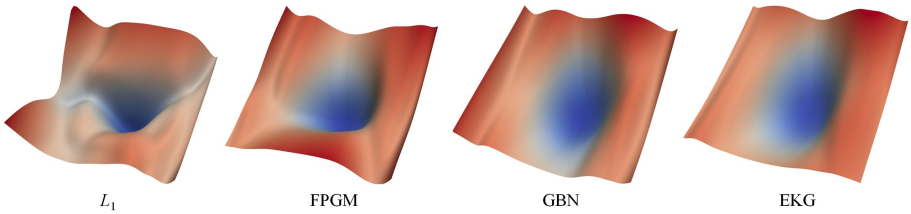

Fig. 4. Visualization of loss landscapes of sub-networks searched by various filter importance scoring algorithms

	Score	L_1 [13]	FPGM [16]	GBN [44]	EKG
CIFAR10	Validation loss	0.1569	0.1066	**0.0710**	0.0713
	Condition number	0.4078	0.1461	0.0414	**0.0211**
	Test accuracy	93.57	93.57	93.70	**93.85**
CIFAR100	Validation loss	1.2050	0.7635	**0.6816**	0.7733
	Condition number	0.2535	0.0838	0.0747	**0.0649**
	Test accuracy	71.43	71.60	71.60	**71.82**

Table 1. Comparison of validation loss, condition number, and test accuracy of various scoring methods on CIFAR10 and CIFAR100

score to each filter using a specific filter scoring technique, found a sub-network with the minimum validation loss, and employed the result for comparison.

First, we visualized the loss landscapes of the sub-networks for each method. As a network for visualization, ResNet-56 trained on CIFAR10 was used. Two directions for constructing the loss landscape were set to a gradient direction and its orthogonal random direction. Fig. 4 is the visualization result. We can observe that the sub-network of EKG has the least fluctuation. In particular, it is noteworthy that L_1-norm-based pruning, which has been adopted in many studies, is in fact not so effective. GBN [44] has the lowest fluctuation among conventional techniques. GBN has a similar structure to the proposed method because it is also based on Taylor expansion. However, since a reward of GBN relies on performance only and does not consider the information of the architecture sufficiently, its loss landscape fluctuation is larger than that of EKG.

Next, let's compare the performance when sub-networks are trained with the same native fine-tuning (see Table 1). We could find that the correlation between the CN and the test accuracy is high. Even though EKG has a higher validation loss than GBN, its potential performance is expected to be higher because the CN or the loss landscape fluctuation of EKG is smaller. Actually, EKG's test accuracy on CIFAR10 was the highest as 93.85%. Therefore, we can find that the proposed search process based on ensemble KD effectively finds the sub-network of smoother loss landscape. Also, the loss landscape fluctuation is a very accurate indicator for evaluating potential performance.

Teacher		ResNet-56			MobileNet-v2		
Search	Fine-tune	CIFAR10	CIFAR100	GPU hours	CIFAR10	CIFAR100	GPU hours
Baseline		93.84	72.62	0.44	94.21	76.07	1.83
None	None	93.78 (±0.07)	71.63 (±0.21)	0.19 / 0.50	93.69 (±0.06)	74.27 (±0.18)	0.31 / 1.35
Single	None	93.54 (±0.09)	71.66 (±0.17)	0.21 / 0.50	93.73 (±0.04)	74.10 (±0.12)	0.35 / 1.35
Ensemble	None	93.85 (±0.10)	71.82 (±0.20)	0.21 / 0.50	93.89 (±0.04)	74.53 (±0.16)	0.35 / 1.35
Ensemble	Single	94.02 (±0.08)	72.62 (±0.15)	0.22 / 0.88	94.44 (±0.09)	76.11 (±0.15)	0.38 / 2.36
Ensemble	Ensemble	**94.09** (±0.07)	**72.93** (±0.16)	0.22 / 0.68	**94.52** (±0.05)	**76.29** (±0.17)	0.38 / 1.64

Table 2. Ablation study to verify each part of the proposed method. Here, None, Single, and Ensemble refer to three methods of using the teacher network. GPU hours were measured in search and fine-tuning phases, respectively

4.2 Ablation study

This section analyzes the effects of EKG on the search and fine-tuning phases. In this experiment, three scenarios of the teacher network are compared: None, Single (when only the pre-trained network is used), and Ensemble (when the proposed ensemble teacher is used). ResNet-56 and MobileNet-v2 trained with CIFAR10 and CIFAR100 were used as datasets, and the FLOPs reduction rate was set to 50%. The experimental results are shown in Table 2.

Method	Acc.	FLOPs ↓	Param ↓
TAS [6]	93.69	52.7	-
ABC [22]	93.23	54.13	52.20
GAL [23]	91.58	60.2	65.9
GBN-60 [44]	93.43	60.1	53.5
FPGM [16]	93.49	52.6	-
Hrank [21]	93.17	50	57.6
LFPC [15]	93.34	52.9	-
DSA [29]	92.93	52.6	-
ManiDP [40]	93.64	62.4	-
NPPM [8]	93.40	50.0	-
SRR-GR [41]	93.75	53.8	-
ResRep [5]	93.71	52.91	-
GDP [12]	93.97	53.35	-
EKG	93.92	55.22	33.69
	93.69	65.11	46.52

Table 3. Comparison with various techniques for ResNet-56 trained on CIFAR10. Here, Acc is the accuracy, and FLOPs ↓ and Param ↓ are the reduction rates of FLOPs and the number of parameters, respectively

First, let's look at the sub-network search phase. Although the search process of 'None' has almost no difference from that of GBN except for the score calculation position, 'None' achieved some performance improvement over GBN. In the case of 'Single,' over-constraint occurs due to the far performance gap between the pre-trained and interim sub-networks. Since interim sub-networks did not undergo fine-tuning to recover the information loss, this phenomenon is further highlighted. Thus, we can observe no performance improvement or rather a decrease in most configurations. For example, in the case of ResNet-56 on CIFAR10, the performance of 'None' and 'Single' was 93.78% and 93.54%, respectively. That is, using knowledge rather causes degradation of 0.24%. On the other hand, if bridging the pre-trained and pruned networks through interim sub-networks, i.e., 'Ensemble,' a sub-network of high potential performance can be selected because of more effective knowledge-based guidance. For example, in the case of ResNet-56 on CIFAR10, the performance of

'Ensemble' is 0.31% higher than 'Single' and outperforms 'None.' Here, it is noteworthy that the performance of 'None' is already comparable than conventional algorithms (see Fig. 5). Therefore, performance improvement of 'Ensemble' is sufficiently acceptable considering that it requires almost no cost.

Next, because the fine-tuning phase re-learns the sub-network, the over-constraints caused by the far performance gap are somewhat mitigated. Accordingly, even if only a single teacher is used, the performance of the pruned network is sufficiently improved, reaching 94.02% in the case of ResNet-56 on CIFAR10. However, as the time for forwarding the teacher network is added, the training time increases by 71.09% compared to 'None.' On the other hand, when training with the proposed memory bank, which ensembles the knowledge of interim sub-networks, that is, in the case of 'Ensemble' knowledge, we can achieve similar performance improvement with only 29.03% increase in training time. This is because the memory bank provides qualified guidance by encapsulating information from many networks. Therefore, each part of the proposed method contributes to effectively providing a lightweight network with high performance at a low cost.

4.3 Performance evaluation

Experiments using ResNet family [14] prove that the proposed method reaches SOTA performance. The datasets used in the experiments are CIFAR10 and ImageNet.

First, Table 3 shows the experimental results for ResNet-56 trained on CIFAR10. For example, when 65.11% of FLOPs are removed by the proposed method, the performance reaches 93.69%. This proves that EKG outperforms the other methods with similar FLOPs reduction rates. To analyze the performance of EKG in detail, Fig. 5 plots the performances at various FLOPs reduction rates. We can observe that EKG provides higher performance than other methods at all FLOPs reduction rates. Note that NN (i.e, 'None'), where any knowledge is not utilized in both phases, already shows a better or comparable performance than most conventional methods. Here, EN utilizing 'Ensemble' knowledge only in the search phase gives sufficient performance improvement. This is because EKG accurately searches for a sub-network with high

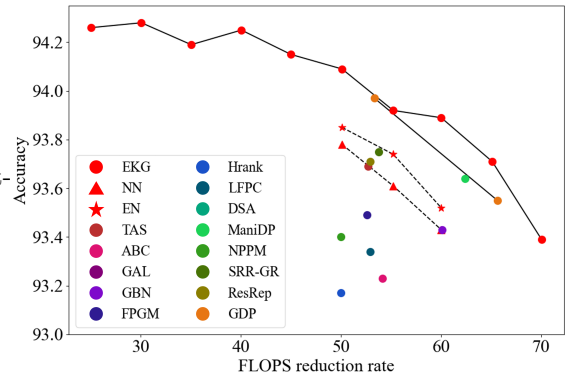

Fig. 5. FLOPS reduction rate-accuracy of various pruning techniques for ResNet-56 on CIFAR10

ResNet	Method	Top-1 (diff.)	Top-5 (diff.)	FLOPs ↓
18	TAS [6]	69.15 (-1.50)	89.19 (-0.68)	33.3
	ABC [22]	67.80 (-1.86)	88.00 (-1.08)	46.9
	FPGM [16]	68.41 (-1.87)	88.48 (-1.15)	41.8
	DSA [29]	68.62 (-1.11)	88.25 (-0.82)	40.0
	DMCP [11]	69.20	N/A	43.0
	ManiDP [40]	68.88 (-0.88)	88.76 (-0.32)	51.0
	EKG	69.39 (-0.99)	88.65 (-0.87)	50.1
34	FPGM [22]	72.63 (-1.28)	91.08 (-0.54)	41.1
	SFP [16]	71.84 (-2.09)	89.70 (-1.92)	41.1
	NPPM [8]	73.01 (-0.29)	91.30 (-0.12)	44.0
	ManiDP [40]	73.30 (-0.01)	91.42 (-0.00)	46.8
	EKG	73.51 (-0.34)	91.27 (-0.19)	45.1
50	DSA [30]	75.1 (-0.92)	92.45 (-0.41)	40.5
	FPGM [22]	75.59 (-0.56)	92.63 (-0.24)	42.7
	BNP [26]	75.51 (-1.01)	92.43 (-0.66)	45.6
	GBN [44]	76.19 (+0.31)	92.83 (-0.16)	41.0
	TAS [6]	76.20 (-1.26)	92.06 (-0.81)	44.1
	SRR-GR [41]	75.76 (-0.37)	92.67 (-0.19)	45.3
	NPPM [8]	75.96 (-0.19)	92.75 (-0.12)	56.2
	ResRep [5]	76.15 (-0.00)	92.89 (+0.02)	54.9
	Autoslim [45]	75.6	N/A	51.6
	DMCP [11]	76.20	N/A	46.7
	CafeNet [36]	76.90	93.3	52.0
	BCNet [37]	76.90	93.3	52.0
	EKG	76.43 (-0.02)	93.13 (-0.02)	45.0
		75.93 (-0.52)	92.82 (-0.33)	55.0
	EKG-BYOL	76.60 (-0.40)	93.23 (-0.31)	55.0

Table 4. Comparison with various techniques for ResNet family trained on ImageNet

potential performance even at high reduction rates. In particular, the higher performance than the pre-trained network even at low FLOPs reduction rates shows that the proposed fine-tuning has a similar regularization effect even though the actual teacher network is not used. Also, this phenomenon indicates that better fine-tuning strategy is more crucial than better sub-network search. Next, Table 4 shows the experimental results for ResNet family trained on ImageNet. EKG achieved SOTA performance in all architectures. For instance, top-1 accuracy of EKG reached 73.51% at 45.1% FLOPs reduction rate for ResNet-34. Since EKG can be plugged-in easily, we adopted a primitive search algorithm, i.e., the greedy search of Autoslim [45] to verify EKG. As a result, EKG improved Autoslim by 0.34% in top-1 accuracy and by 3.5% in FLOPs reduction rate for ResNet-50. On the other hand, in the case of ResNet-50, EKG showed lower performance than the latest supernet-based algorithms, i.e., CafeNet and BCNet. However, since EKG can employ well-trained networks which are not allowed for supernet-based algorithms, it can overcome even CafeNet and BCNet. For example, when using BYOL [10], which is the best among available Tensorflow-based ResNet-50, the top-1 accuracy of EKG was improved up to 76.6% without cost increase. Therefore, if a better pre-trained network is released, EKG will be able to overwhelm the supernet in performance.

For further analysis, the GPU hours-accuracy plot is given in Fig. 6, together with the results for ResNet-50. The cost of pruning and fine-tuning ResNet-50 with EKG is about 315 GPU hours, which is much lower than the other meth-

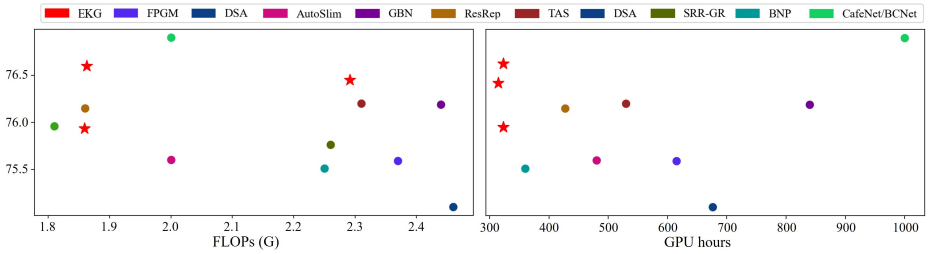

Fig. 6. Performance analysis for ResNet-50 trained on ImageNet. The left plot is the FLOPs reduction rate-Top-1 accuracy, and the right plot is the GPU hours-Top-1 accuracy

ods with similar performance. In particular, EKG is computationally efficient because we used just mid-range GPUs mounted on general-purpose workstations. In other words, EKG democratizes DNNs because it allows more users to learn high-performance lightweight DNNs.

4.4 Limitation

EKG improves the performance of conventional NAS-based pruning by effectively searching and fine-tuning sub-networks. However, some limitations still remain. First, as mentioned above, CN looks impractical because it cannot directly replace performance due to computational complexity. In this paper, we indirectly induced the sub-network of the smoother loss landscape to be selected, but additional research is needed on how the CN can be used as a direct metric. Second, the fact that EKG has not been evaluated in sufficiently many configurations obscures its generalization a little bit. In order to analyze the performance of EKG more clearly, it is necessary to verify light-weight architectures on a large dataset or to attach to other search algorithms, e.g., evolutionary search.

5 Conclusion

This paper proposes EKG as an efficient and effective solution for search reward and fine-tuning strategy, which have been rarely considered in existing NAS-based pruning approaches. In particular, sub-network evaluation based on loss landscape fluctuations can reflect generality more accurately than validation performance which may cause a high risk of overfitting. Furthermore, EKG is pretty valuable because it can be easily plugged-in to the existing NAS-based pruning. Despite some unresolved limitations, we expect this paper to provide meaningful insight to filter pruning researchers. In particular, we pointed out and corrected the fact that inaccurate validation performance has been adopted without any doubt.

References

1. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al.: Tensorflow: A system for large-scale machine learning. In: 12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16). pp. 265–283 (2016)
2. Cai, H., Gan, C., Wang, T., Zhang, Z., Han, S.: Once-for-all: Train one network and specialize it for efficient deployment. In: International Conference on Learning Representations (2019)
3. Chen, T., Zhang, Z., Liu, S., Chang, S., Wang, Z.: Robust overfitting may be mitigated by properly learned smoothening. In: International Conference on Learning Representations (2021), https://openreview.net/forum?id=qZzy5urZw9
4. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations (2020)
5. Ding, X., Hao, T., Tan, J., Liu, J., Han, J., Guo, Y., Ding, G.: Resrep: Lossless cnn pruning via decoupling remembering and forgetting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4510–4520 (2021)
6. Dong, X., Yang, Y.: Network pruning via transformable architecture search. In: Advances in Neural Information Processing Systems. pp. 760–771 (2019)
7. Frankle, J., Carbin, M.: The lottery ticket hypothesis: Finding sparse, trainable neural networks. In: International Conference on Learning Representations (2018)
8. Gao, S., Huang, F., Cai, W., Huang, H.: Network pruning via performance maximization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9270–9280 (2021)
9. Gotmare, A., Keskar, N.S., Xiong, C., Socher, R.: A closer look at deep learning heuristics: Learning rate restarts, warmup and distillation. arXiv preprint arXiv:1810.13243 (2018)
10. Grill, J.B., Strub, F., Altché, F., Tallec, C., Richemond, P.H., Buchatskaya, E., Doersch, C., Pires, B.A., Guo, Z.D., Azar, M.G., et al.: Bootstrap your own latent: A new approach to self-supervised learning. arXiv preprint arXiv:2006.07733 (2020)
11. Guo, S., Wang, Y., Li, Q., Yan, J.: Dmcp: Differentiable markov channel pruning for neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1539–1547 (2020)
12. Guo, Y., Yuan, H., Tan, J., Wang, Z., Yang, S., Liu, J.: Gdp: Stabilized neural network pruning via gates with differentiable polarization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5239–5250 (2021)
13. Han, S., Mao, H., Dally, W.J.: Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. arXiv preprint arXiv:1510.00149 (2015)
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
15. He, Y., Ding, Y., Liu, P., Zhu, L., Zhang, H., Yang, Y.: Learning filter pruning criteria for deep convolutional neural networks acceleration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
16. He, Y., Liu, P., Wang, Z., Hu, Z., Yang, Y.: Filter pruning via geometric median for deep convolutional neural networks acceleration. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4340–4349 (2019)

17. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015)
18. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Tech. rep., Citeseer (2009)
19. Lee, S., Heo, B., Ha, J.W., Song, B.C.: Filter pruning and re-initialization via latent space clustering. IEEE Access **8**, 189587–189597 (2020). https://doi.org/10.1109/ACCESS.2020.3031031
20. Li, H., Xu, Z., Taylor, G., Studer, C., Goldstein, T.: Visualizing the loss landscape of neural nets. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems. pp. 6391–6401 (2018)
21. Lin, M., Ji, R., Wang, Y., Zhang, Y., Zhang, B., Tian, Y., Shao, L.: Hrank: Filter pruning using high-rank feature map. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
22. Lin, M., Ji, R., Zhang, Y., Zhang, B., Wu, Y., Tian, Y.: Channel pruning via automatic structure search. arXiv preprint arXiv:2001.08565 (2020)
23. Lin, S., Ji, R., Yan, C., Zhang, B., Cao, L., Ye, Q., Huang, F., Doermann, D.: Towards optimal structured cnn pruning via generative adversarial learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2790–2799 (2019)
24. Liu, Z., Mu, H., Zhang, X., Guo, Z., Yang, X., Cheng, K.T., Sun, J.: Metapruning: Meta learning for automatic neural network channel pruning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3296–3305 (2019)
25. Lu, X., Huang, H., Dong, W., Li, X., Shi, G.: Beyond network pruning: a joint search-and-training approach. In: IJCAI. pp. 2583–2590 (2020)
26. Lu, X., Huang, H., Dong, W., Li, X., Shi, G.: Beyond network pruning: a joint search-and-training approach. In: IJCAI. pp. 2583–2590 (2020)
27. Mirzadeh, S.I., Farajtabar, M., Li, A., Levine, N., Matsukawa, A., Ghasemzadeh, H.: Improved knowledge distillation via teacher assistant. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 5191–5198 (2020)
28. Molchanov, P., Tyree, S., Karras, T., Aila, T., Kautz, J.: Pruning convolutional neural networks for resource efficient inference. In: International Conference on Learning Representations (2017)
29. Ning, X., Zhao, T., Li, W., Lei, P., Wang, Y., Yang, H.: Dsa: More efficient budgeted pruning via differentiable sparsity allocation. In: Proceedings of the European Conference on Computer Vision (ECCV) (2020)
30. Ning, X., Zhao, T., Li, W., Lei, P., Wang, Y., Yang, H.: Dsa: More efficient budgeted pruning via differentiable sparsity allocation. arXiv preprint arXiv:2004.02164 (2020)
31. Nocedal, J., Wright, S.: Numerical optimization. Springer Science & Business Media (2006)
32. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV) **115**(3), 211–252 (2015). https://doi.org/10.1007/s11263-015-0816-y
33. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4510–4520 (2018)
34. Santurkar, S., Tsipras, D., Ilyas, A., Madry, A.: How does batch normalization help optimization? In: Proceedings of the 32nd international conference on neural information processing systems. pp. 2488–2498 (2018)

35. Soper, H., Young, A., Cave, B., Lee, A., Pearson, K.: On the distribution of the correlation coefficient in small samples. appendix ii to the papers of" student" and ra fisher. Biometrika **11**(4), 328–413 (1917)
36. Su, X., You, S., Huang, T., Wang, F., Qian, C., Zhang, C., Xu, C.: Locally free weight sharing for network width search. In: International Conference on Learning Representations (2021), https://openreview.net/forum?id=S0UdquAnr9k
37. Su, X., You, S., Wang, F., Qian, C., Zhang, C., Xu, C.: Bcnet: Searching for network width with bilaterally coupled network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2175–2184 (June 2021)
38. Tan, X., Ren, Y., He, D., Qin, T., Liu, T.Y.: Multilingual neural machine translation with knowledge distillation. In: International Conference on Learning Representations (2019), https://openreview.net/forum?id=S1gUsoR9YX
39. Tang, J., Shivanna, R., Zhao, Z., Lin, D., Singh, A., Chi, E.H., Jain, S.: Understanding and improving knowledge distillation. arXiv preprint arXiv:2002.03532 (2020)
40. Tang, Y., Wang, Y., Xu, Y., Deng, Y., Xu, C., Tao, D., Xu, C.: Manifold regularized dynamic network pruning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5018–5028 (2021)
41. Wang, Z., Li, C., Wang, X.: Convolutional neural network pruning with structural redundancy reduction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14913–14922 (2021)
42. Wu, Y., et al.: Tensorpack. https://github.com/tensorpack/ (2016)
43. Yalniz, I.Z., Jégou, H., Chen, K., Paluri, M., Mahajan, D.: Billion-scale semi-supervised learning for image classification (2019)
44. You, Z., Yan, K., Ye, J., Ma, M., Wang, P.: Gate decorator: Global filter pruning method for accelerating deep convolutional neural networks. In: Advances in Neural Information Processing Systems. pp. 2133–2144 (2019)
45. Yu, J., Huang, T.: Autoslim: Towards one-shot architecture search for channel numbers. arXiv preprint arXiv:1903.11728 (2019)
46. Yu, J., Huang, T.S.: Universally slimmable networks and improved training techniques. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1803–1811 (2019)
47. Yu, J., Yang, L., Xu, N., Yang, J., Huang, T.: Slimmable neural networks (2018)