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

Seunghyun Lee*, Byung Cheol Song
Department of Electrical and Computer Engineering, Inha University, Republic of Korea

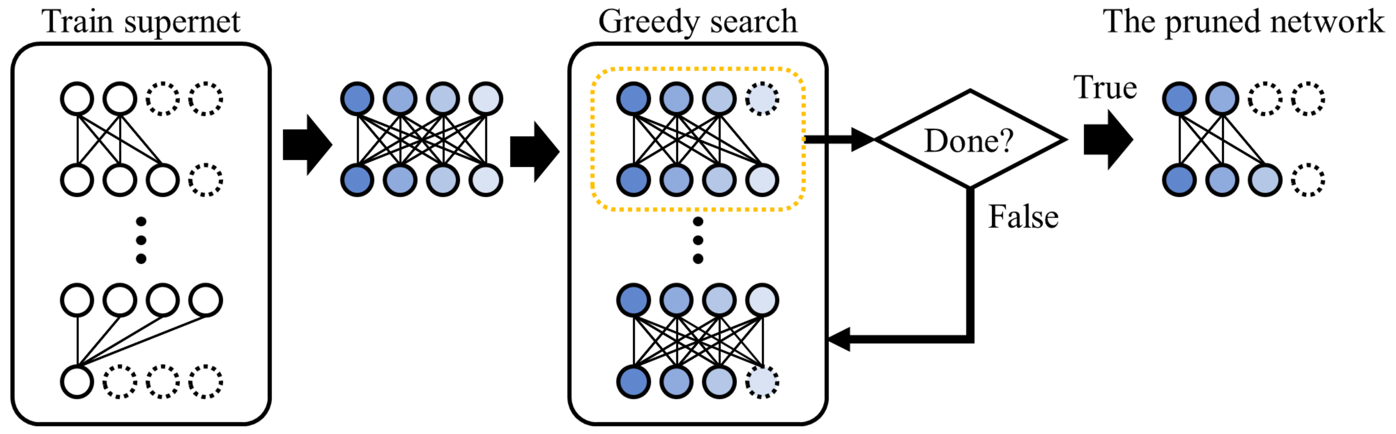
ECCV TEL AVIV 2022

Introduction

Search-based filter pruning

- Search the best sub-network with a given FLOPS
- **Supernet**:

  Network trained to have sorted filters according to importance



Problem definition

- **(1)** Validation loss does not represent its potential
- **(2)** Fine-tuning for the pruned network is essential

  but has not been researched so far

Key ideas

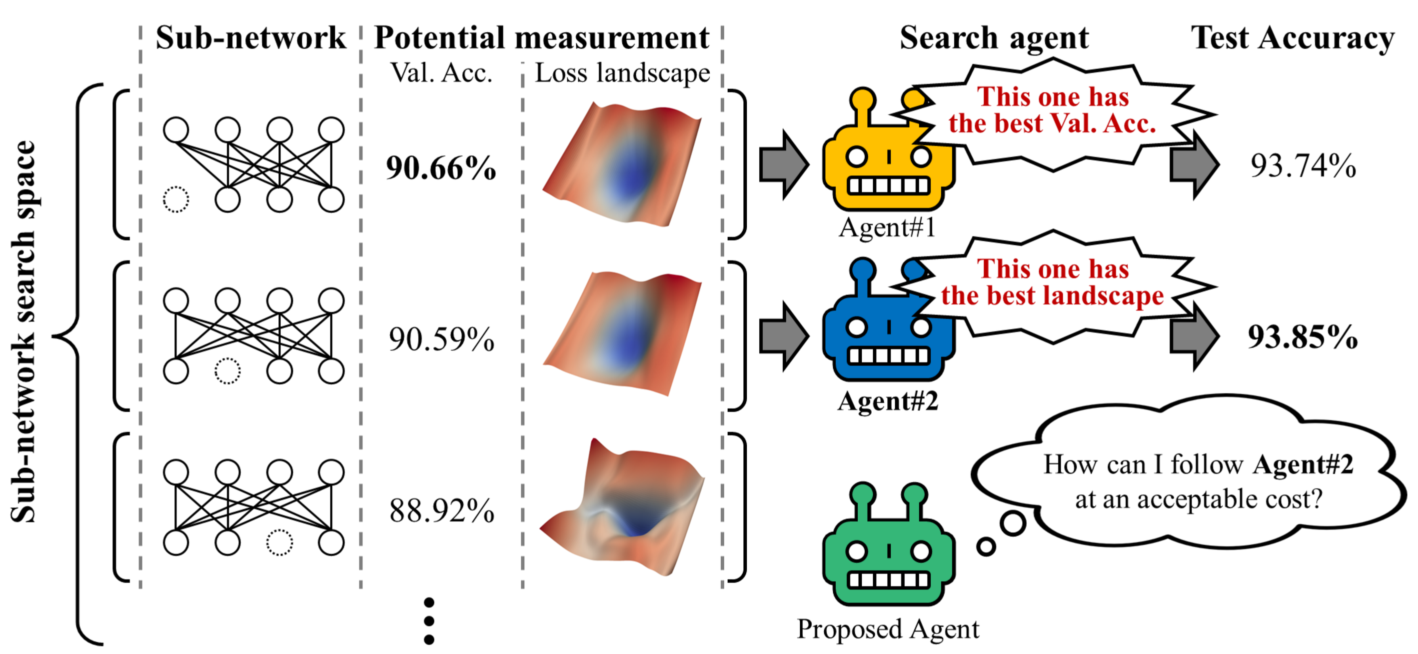
- **(1)** Define a novel measurement to evaluate sub-network

  → Loss landscape smoothness

- **(2)** Distill knowledge from the search phase and

  utlize it in both search and fine-tuning phase

  → Indirect solution to achieve **(1)**



Advantages

- EKG needs only **negligible computational cost**

  → SOTA performance using acceptable GPU hours

Methods

(1) Accurate potential performance evaluation

- The higher generalization, **The smoother loss landscape**
- Hessian requires to check smoothness, **which is too costy**

  → **Knowledge distillation** as an indirect solution

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

(2) Search with ensemble knowledge guidance

- Define interim sub-networks, i.e., by-products of search as teacher networks
- Store and ensemble teacher network's knowledge into memory bank

  → Give a **gentle guidance by ensemble knowledge** with a negligible cost

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

(2) Fine-tuning with ensemble knowledge guidance

- Sample some of interim sub-networks from the sub-network search phase

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

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

- Contrastive learning strategy is adopted to prevent overfitting

  → Make augmented features clustered into the fixed teacher knowledge

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

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

Overall algorithm

Algorithm 1: The proposed pruning algorithm

Input : $\Theta_0, \mathcal{D}^{train}, r$ **Output** : Fine-tuned network

1: Sample $\mathcal{D}^{subset}$ and $\mathcal{D}^{val}$ in $\mathcal{D}^{train}$ and fine-tune Θ_0 in an epoch on $\mathcal{D}^{subset}$
2: Store initial ensemble knowledge $\mathcal{T}_0$

search phase

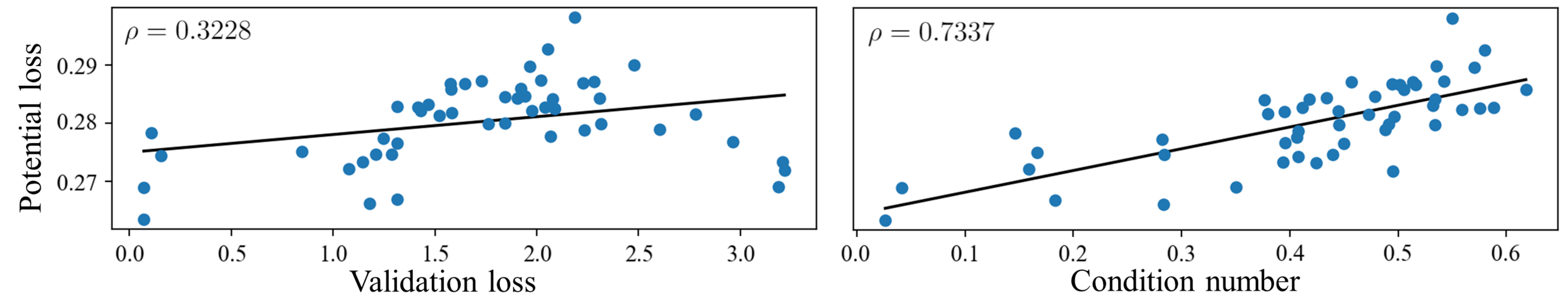
3: **Repeat**
4: Compute filter importance scores $\mathcal{S}(\theta)$ by Eq. (9)
5: Sample candidates Φ_i in each layers with r.
6: Select $\phi^*_{i,l}$ by Eq. (4) that maximizes Eq. (7).
7: Get next pruned network Θ_{i+1} by Eq. (3).
8: Update ensemble knowledge $\mathcal{T}_{i+1}$ by Eq. (8)
9: $i = i + 1$
10: **Until** FLOPs reduction rate reaches the goal
11: $\Theta^* = \Theta_i$
12: Build memory bank by Eq. (10)

fine-tuning phase

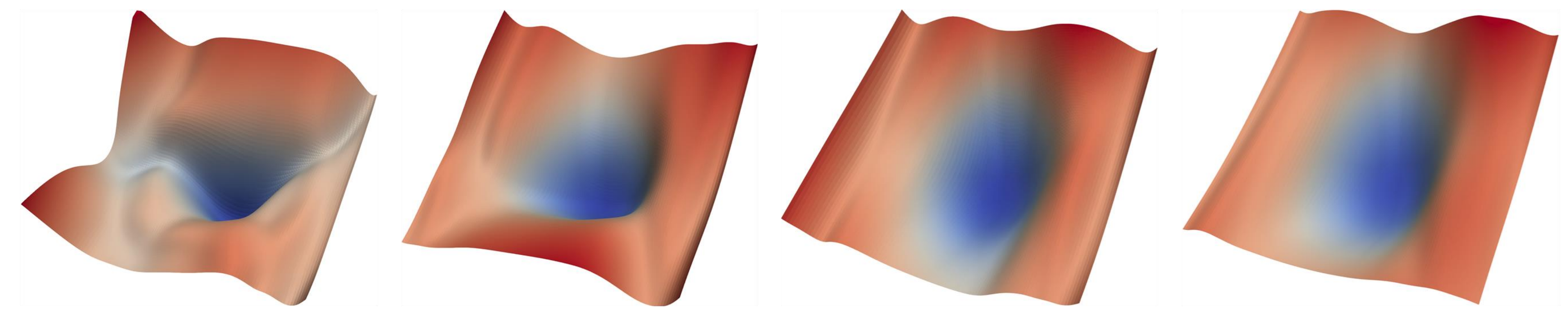
13: **Repeat**
14: Get training sample in $\mathcal{D}^{train}$ and apply two augmentation functions.
15: Minimize loss function in Eq. (12).
16: **Until** Training is done

Experimental results

Feasibility check for the loss landscape smoothness



Comparison of potential loss vs validation loss and condition number



L_1 FPGM GBN EKG

	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**

Comparison of EKG and conventional search manner

Ablation study

Teacher Search	Teacher Fine-tune	ResNet-56 CIFAR10	ResNet-56 CIFAR100	ResNet-56 GPU hours	MobileNet-v2 CIFAR10	MobileNet-v2 CIFAR100	MobileNet-v2 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

Comparison of the way to utilize knowledge in each phase

Performance comparison of ResNet-family trained on ImageNet

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