Practical Network Acceleration with Tiny Sets

Guo-Hua Wang Jianxin Wu

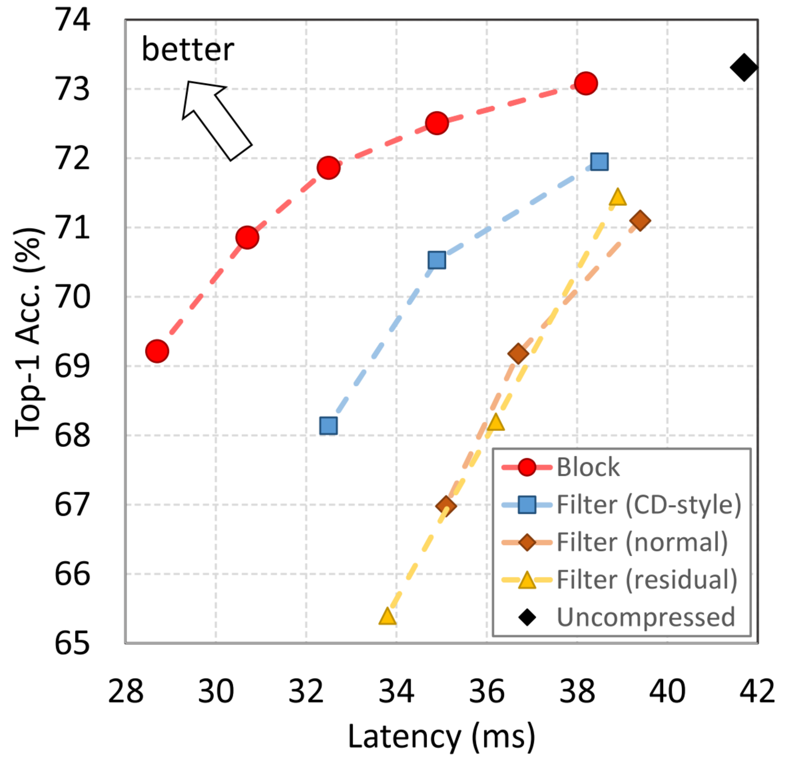
State Key Laboratory for Novel Software Technology, Nanjing University

Paper Code

JUNE 18-22, 2023 CVPR VANCOUVER, CANADA

1. Introduction

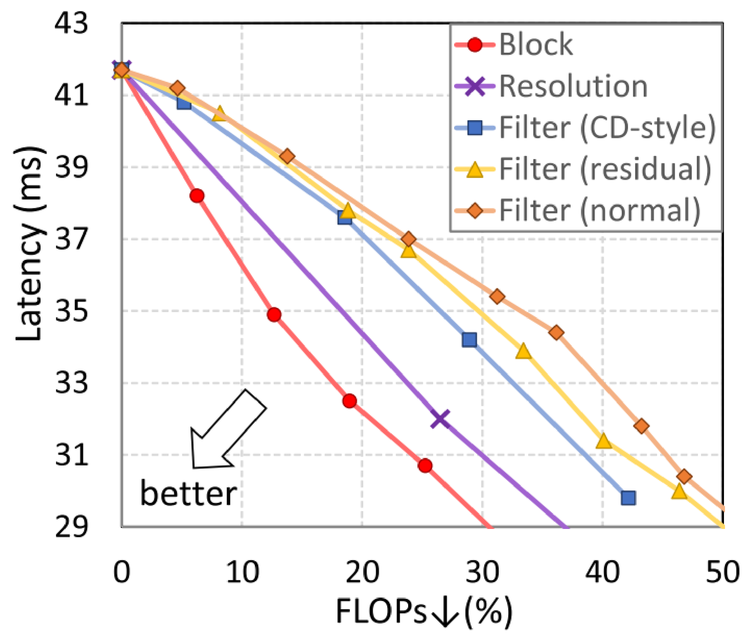
- In this paper, we propose an algorithm named PRACTISE to accelerate deep networks with tiny training sets.
  - PRACTISE: practical network acceleration with tiny sets
  - Deep networks: ResNet, MobileNet, etc
  - Tiny training sets: 50~1000 images
- Key property
  - High latency-accuracy performance
  - Small training sets
  - Robust to out-of-domain images
  - Fast training (1.5 hours)



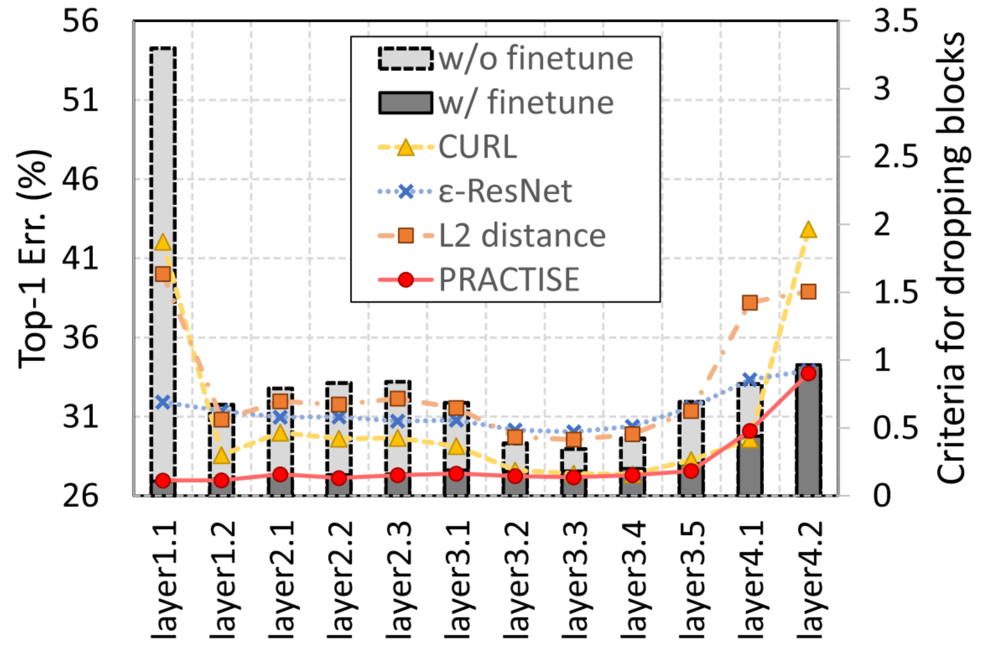
2. Motivation

- Latency-accuracy vs. FLOPs-accuracy: more practical
- Drop blocks: higher acceleration ratio, more convex optimization
- Recoverability: predict the finetuned acc. to prune blocks

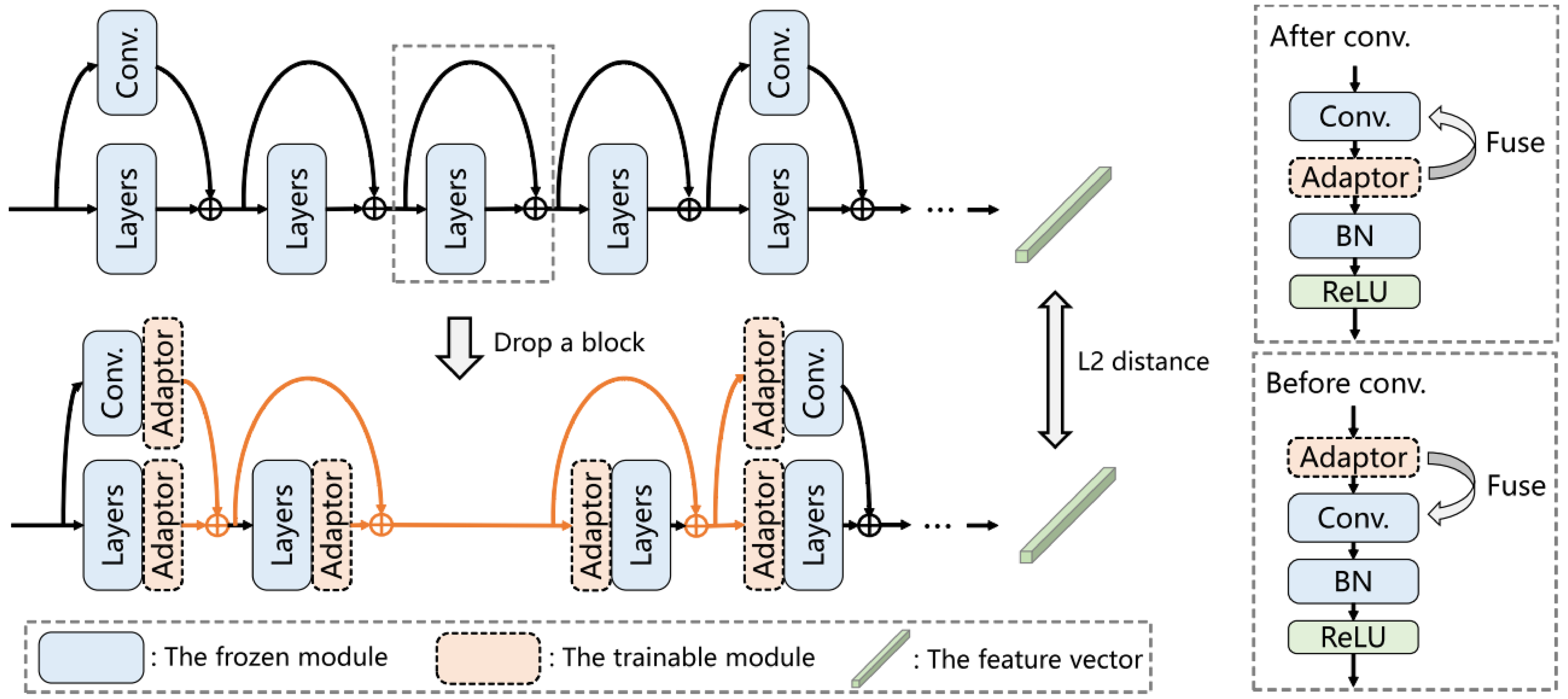
High acceleration ratio



Different criteria for dropping blocks



3. Method



Algorithm 1: PRACTISE

Input: The original model $\mathcal{M}_O$, the number of dropped blocks k, the tiny training data $\mathcal{D}_\mathcal{T}$

Test the latency of $\mathcal{M}_O$;

for *each block* $\mathcal{B}_i$ **do**

- Drop $\mathcal{B}_i$ to obtain the pruned model $\mathcal{M}_{P(\mathcal{B}_i)}$;
- Test latency of $\mathcal{M}_{P(\mathcal{B}_i)}$ and find $\tau(\mathcal{B}_i)$ (Eq. 2);
- Insert adaptors;
- Compute $\mathcal{R}(\mathcal{B}_i)$ with $\mathcal{D}_\mathcal{T}$ (Eq. 1);
- Compute the score $s(\mathcal{B}_i)$ (Eq. 3);
- Add $\mathcal{B}_i$ back and remove all adaptors;

Choose the top k blocks with the minimum scores;

Drop these k blocks to obtain $\mathcal{M}_P$;

Finetune $\mathcal{M}_P$ with $\mathcal{D}_\mathcal{T}$ by minimizing $\mathcal{L}$ (Eq. 4);

return The pruned model $\mathcal{M}_P$

$$\mathcal{R}(\mathcal{B}_i) = \min_{\alpha} \mathbb{E}_{x\sim p(x)} \|\mathcal{M}_O(x;\theta) - \mathcal{M}_{P(\mathcal{B}_i)}(x;\theta\backslash b_i, \alpha)\|_F^2 \quad (1)$$

$$\tau(\mathcal{B}_i) = \frac{lat_{\mathcal{M}_O} - lat_{\mathcal{M}_{P(\mathcal{B}_i)}}}{lat_{\mathcal{M}_O}}, \quad (2)$$

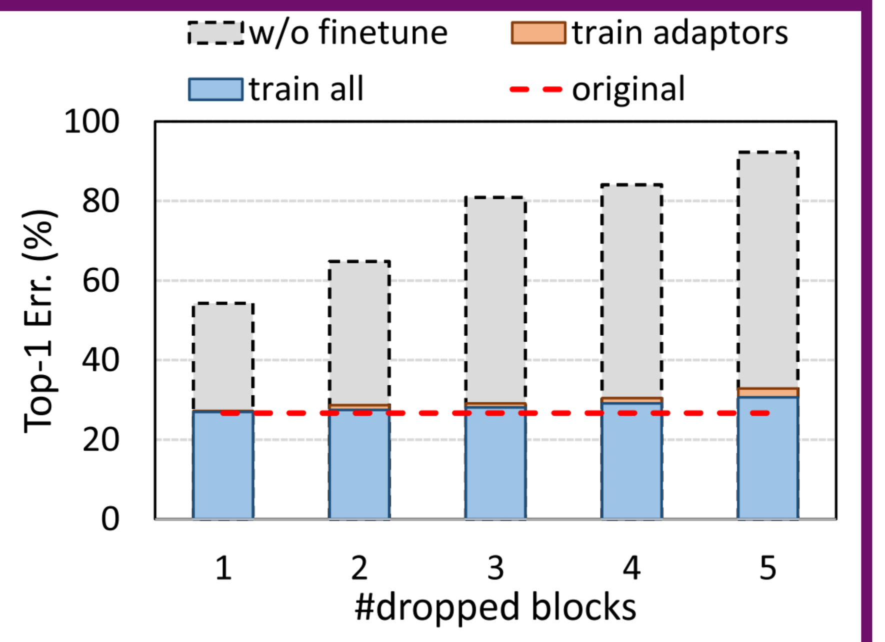
$$s(\mathcal{B}_i) = \frac{\mathcal{R}(\mathcal{B}_i)}{\tau(\mathcal{B}_i)}. \quad (3)$$

$$\mathcal{L} = \|\mathcal{M}_O(x;\theta_O) - \mathcal{M}_P(x;\theta_P)\|_F^2, \quad (4)$$

4. Experiments

Different criteria for dropping blocks

#Dropped blocks	1	2	3	4	5
Random	71.12	71.27	67.89	64.27	63.47
CURL [21]	72.33	71.08	69.14	65.48	64.97
ϵ-ResNet [34]	72.51	71.20	69.00	67.90	64.75
L2 distance	72.51	71.20	69.00	68.61	64.93
PRACTISE	**73.02**	**72.52**	**71.92**	**70.86**	**69.35**



ImageNet results

Method	Latency (ms)	50	100	500	1000
BP (filter)	33.8 (18.9% ↓)	$24.2_{\pm0.92}/52.7_{\pm1.36}$	$27.6_{\pm0.41}/56.7_{\pm0.62}$	$42.9_{\pm0.28}/70.5_{\pm0.27}$	$51.2_{\pm0.32}/76.5_{\pm0.16}$
BP (block)	**32.5** (**22.1**% ↓)	$\mathbf{60.6}_{\pm0.62}/\mathbf{83.5}_{\pm0.42}$	$\mathbf{61.6}_{\pm0.31}/\mathbf{84.3}_{\pm0.36}$	$\mathbf{65.0}_{\pm0.19}/\mathbf{86.5}_{\pm0.20}$	$\mathbf{66.8}_{\pm0.18}/\mathbf{87.5}_{\pm0.13}$
KD [10]	33.8 (18.9% ↓)	$30.1_{\pm0.69}/57.7_{\pm1.10}$	$33.1_{\pm0.43}/61.0_{\pm0.53}$	$45.7_{\pm0.26}/72.2_{\pm0.25}$	$50.5_{\pm0.29}/75.9_{\pm0.23}$
FSKD [12]	33.8 (18.9% ↓)	$31.1_{\pm0.90}/56.5_{\pm1.10}$	$36.6_{\pm0.44}/63.1_{\pm0.46}$	$42.8_{\pm0.49}/69.1_{\pm0.58}$	$44.9_{\pm0.20}/70.5_{\pm0.29}$
MiR [30]	33.8 (18.9% ↓)	$59.9_{\pm0.30}/83.2_{\pm0.31}$	$62.1_{\pm0.22}/84.8_{\pm0.18}$	$65.4_{\pm0.07}/87.0_{\pm0.03}$	$66.6_{\pm0.05}/87.7_{\pm0.04}$
PRACTISE	**32.5** (**22.1**% ↓)	$\mathbf{68.0}_{\pm1.36}/\mathbf{88.2}_{\pm0.77}$	$\mathbf{70.4}_{\pm0.42}/\mathbf{89.7}_{\pm0.23}$	$\mathbf{71.8}_{\pm0.07}/\mathbf{90.5}_{\pm0.02}$	$\mathbf{71.9}_{\pm0.05}/\mathbf{90.6}_{\pm0.04}$

5. Contributions & Conclusions

- Argue that the FLOPs-accuracy tradeoff is a misleading metric for few-shot compression, and advocate that the **latency-accuracy** tradeoff is more crucial in practice.
- The first to reveal **dropping blocks** great potential in few-shot compression.
- Propose a new concept **recoverability** to measure the difficulty of recovering each block, and in determining the priority to drop blocks.
- Propose **PRACTISE**, an algorithm for accelerating networks with tiny sets of images.
- The extraordinary performance: For 22.1% latency reduction, PRACTISE surpasses the previous state-of-the-art (SOTA) method on average by 7.0%.