APPENDIX: WINOGRAD STRUCTURED PRUNING

Anonymous authors
Paper under double-blind review

A LIMITATION OF MIDDLE-GRAINED PRUNING ON WC

The Winograd transformation process converts sparse matrices into dense matrices thus it is challenging to apply Winograd convolution (*WC*) and *pruning* techniques, simultaneously (Liu et al., 2018). This is because the Filter Transformation (FTrans) process densely changes the sparsity of pruned model by *pruning*. FTrans is performed in units of 2D tensor ($\mathbb{R}^{k_h \times k_w}$) out of 4D tensor ($\mathbb{R}^{n \times c \times k_h \times k_w}$) of the convolution layer. Therefore, *pruning* with a pruning unit size of 2D tensor ($\mathbb{R}^{k_h \times k_w}$) or larger can be compatible simultaneously with *WC*. Filter pruning (FP), which is pruned in units of 3D tensor ($\mathbb{R}^{c \times k_h \times k_w}$), can be compatible with *WC* (Yu et al., 2019). Middle-grained pruning (e.g., 1×n Pruning (Lin et al., 2022) and Block Sparse (Narang et al., 2017)) with a pruning unit size of more than 2D tensor ($\mathbb{R}^{k_h \times k_w}$) can also be compatible with *WC*. As shown in Figure 1, the middle-grained pruning prunes in units of vector or block when the convolution layer is converted to matrix multiplication using im2col (called lowering method) (Chellapilla et al., 2006; Chetlur et al., 2014). When the middle-level pruned model is converted to col2im, most of them are pruned in 2D tensor units ($\mathbb{R}^{k_h \times k_w}$). Therefore, middle-grained pruning also be compatible with *WC*, like FP. However, unlike FP, the middle-grained pruned model have unstructured data pattern in Winograd-domain. As a result, middle-grained pruning is still difficult to improve performance like FP or our proposed ABWSP without an appropriate GPU library.

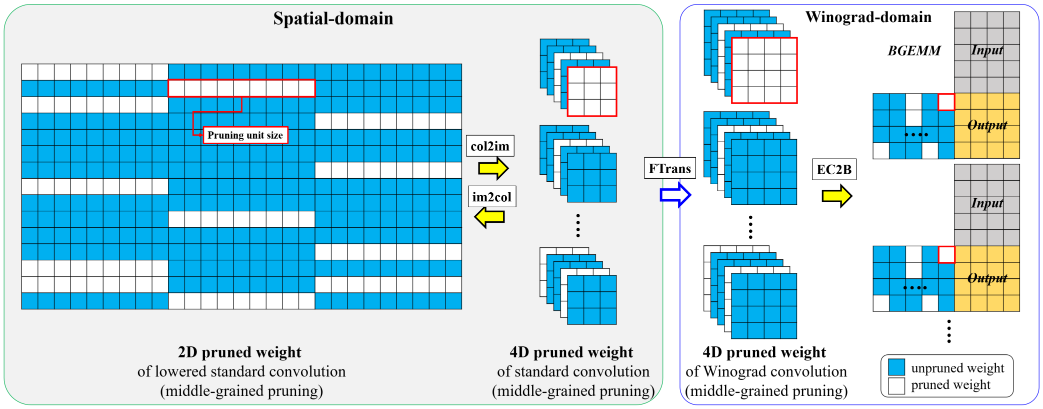


Figure 1: Overview of middle-grained pruning on *WC* and standard convolution

B LARGE PRUNING UNIT SIZE PROBLEM OF FP

B.1 PRUNING UNIT SIZE ON VGG-16

When the size of the pruning unit increases, there is a notable decrease in the representation power of the network. This is due to the fact that the pruning unit size plays a crucial role in determining the precision of the *pruning* process. As shown in Figure 2, in the case of VGG-16, FP ($p^2 \times \mathbb{R}^{1 \times c}$) has a larger pruning unit size than WSP-R ($\mathbb{R}^{1 \times c}$) in all layers except for the first convolution layer. For example, on VGG-16, the pruning unit sizes of FP on Winograd-domain are either $1,024$, $2,048$, $4,096$, and $8,192$ parameters depending on the layers, except the first convolution layer. On the other hand, the pruning unit size of WSP-R is almost 16 (p^2) times smaller than FP's, because the pruning unit size of WSP-R are either 64, 128, 256, and 512 parameters depending on the layers.

Therefore, our proposed WSP and ABWSP in VGG-16 are more sophisticated *pruning* approaches than FP in terms of accuracy and inference speed.

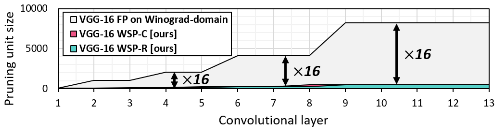


Figure 2: Comparison of pruning unit size of pruned model on VGG-16.

B.2 Comparison of accuracy and pruning ratio

ResNet-18 on ImageNet, Fine-tuning epoch is 40 The side-effect of FP has representation loss when the models are aggressively pruned. As shown in Figure 3, the pruning unit size of FP on the Winograd-domain is $p^2 \times \mathbb{R}^{1\times c}$, which tends to be too large, so it removes the important weights with redundant weights together. In Figure 3, when evaluating FPs with PRs of 30% in ResNet-18, the Top-1 accuracy is reduced by 3%. In 40 fine-tuning epoch, both WWP and WSP have an Top-1 accuracy drop of less than 3% in 90% PR. In particular, FP has an Top-1 accuracy drop of more than 3% in most PRs. Even though FP keeps the matrices in a structured form to leverage the full computing capabilities of existing general-purpose computing devices, FP shows significant accuracy drops in the high PR even with retaining.

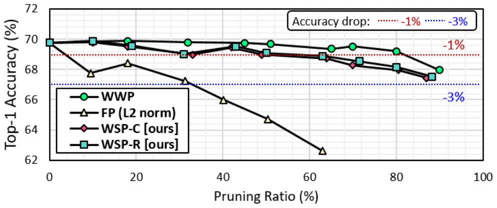


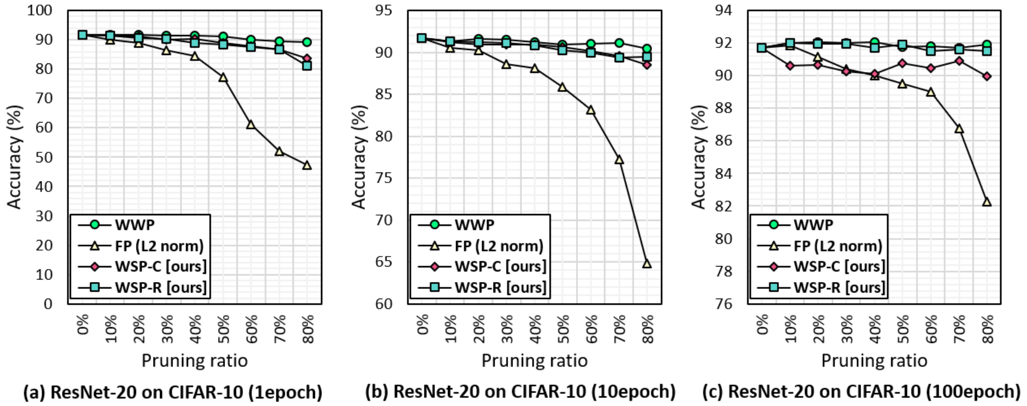
Figure 3: Comparison of accuracy and PR. Top-1 validation and PR for four model on a variation of ResNet-18 on ImageNet. The fine-tuning epoch is 40.

C Effect of Fine-tuning

We evaluate the accuracy according to the variation of fine-tuning epoch with four different *pruning* on ResNet-20 and VGG-16 in CIFAR-10 (see Figure 4 and 5).

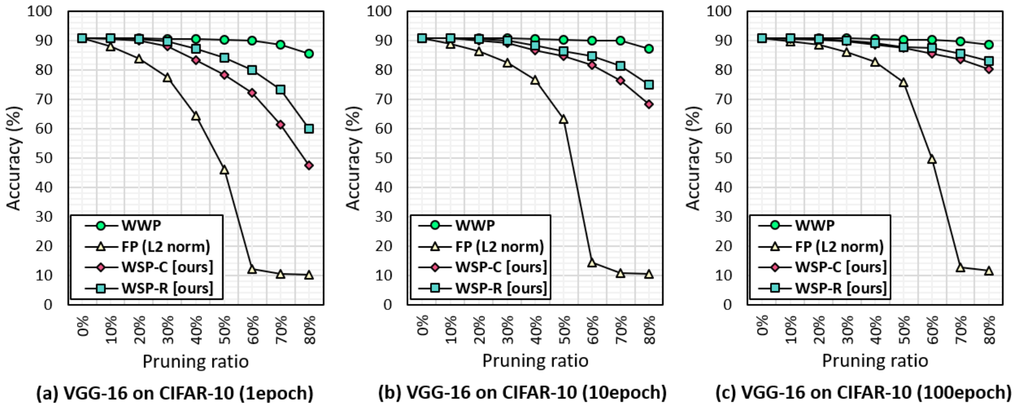
ResNet-20 on CIFAR-10 Since WWP is fine-grained pruning, when epoch is 10 and 100, only a minimum accuracy drop (less than 0.64% and 0.03%) is observed even with 70% of pruning ratio (PR). In Winograd-domain, FP is a method of removing more than hundreds of elements from ResNet-20, so even if the PR is 10% when fine-tuning epoch is 1 and 10, significant accuracy degradation (at least 1.65%) is observed. Even if the FP pruned model is fine-tuned more than 100 epochs, if the PR exceeds 40%, the accuracy drop exceeds 1.72%. Since the pruning unit size of WSP is p^2 times less than FP's, WSP can prune more sophisticated than FP. With only 10 fine-tuning epochs, our proposed WSP shows an accuracy drop of less than 1.5% even with a higher than 60% of PR.

VGG-16 on CIFAR-10 Since WWP is fine-grained pruning, when epoch is 10, only a minimum accuracy drop (less than 0.6%) is observed even with 70% PR. On the other hand, FP is a method of removing thousands of elements from VGG-16, so even if the PR is 20% when fine-tuning epoch is 1 and 10, Significant accuracy degradation (at least 5%) is observed. Even if the FP pruned model is fine-tuned more than 100 epochs, if the PR exceeds 30%, the accuracy drop exceeds 5%. Since the pruning unit size of WSP is $\mathbb{R}^{n\times 1}$ or $\mathbb{R}^{1\times c}$, WSP can prune more sophisticated than FP.



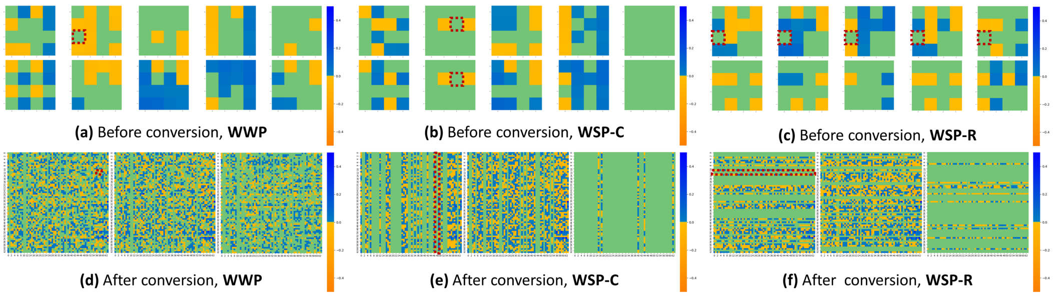
(a) ResNet-20 on CIFAR-10 (1epoch) (b) ResNet-20 on CIFAR-10 (10epoch) (c) ResNet-20 on CIFAR-10 (100epoch)

Figure 4: Comparing WSP with WWP and FP of ResNet-20 on CIFAR-10. We experiment with PR and accuracy using three epochs of fine-tuning: 1, 10, and 100.



(a) VGG-16 on CIFAR-10 (1epoch) (b) VGG-16 on CIFAR-10 (10epoch) (c) VGG-16 on CIFAR-10 (100epoch)

Figure 5: Comparing WSP with WWP and FP of VGG-16 on CIFAR-10. We experiment with PR and accuracy using three epochs of fine-tuning: 1, 10, and 100.



(a) Before conversion, **WWP** (b) Before conversion, **WSP-C** (c) Before conversion, **WSP-R**

(d) After conversion, **WWP** (e) After conversion, **WSP-C** (f) After conversion, **WSP-R**

Figure 6: Pruned filter visualizations in res2a_2a layer of ResNet-18 on ImageNet. Positive, negative and pruned weights are in blue, yellow and green respectively. Dotted redline denotes a pruning unit.

D Pruned Filter Visualization

We use a visualization method to understand that WSP has a regular data pattern and middle pruning unit size at WC. In Figure 6, we sequentially visualized WWP (Liu et al., 2018), WSP-C, and WSP-R. The pruned filter visualization is done with the PR of 50% within the 1% variance. For visualization, we convert the EWMM 4D weight matrix of the WC to BGEMM weight matrix using EC2B converting method. The weight map of WWP shows irregular data patterns in both non-converted and converted weight matrices as shown in Figure 6(a) and 6(d). WSP-C has the same pruning mask between matrices with the same output channel as shown in Figure 6(b). In Figure 6(e), the WSP-C shows regular data patterns which have column-wise vector pruning unit at converted weight. In Figure 6(c), WSP-R has the same pruning mask between matrices with the same input channel. WSP-R has row-wise vector pruned data pattern as shown in Figure 6(f).

REFERENCES

Kumar Chellapilla, Sidd Puri, and Patrice Simard. High performance convolutional neural networks for document processing. 2006.

Sharan Chetlur, Cliff Woolley, Philippe Vandermersch, Jonathan Cohen, John Tran, Bryan Catanzaro, and Evan Shelhamer. cudnn: Efficient primitives for deep learning. *arXiv preprint arXiv:1410.0759*, 2014.

Mingbao Lin, Yuxin Zhang, Yuchao Li, Bohong Chen, Fei Chao, Mengdi Wang, Shen Li, Yonghong Tian, and Rongrong Ji. 1xn pattern for pruning convolutional neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

Xingyu Liu, Jeff Pool, Song Han, and William J Dally. Efficient sparse-winograd convolutional neural networks. In *International Conference on Learning Representations*, 2018.

Sharan Narang, Eric Undersander, and Gregory Diamos. Block-sparse recurrent neural networks. *arXiv preprint arXiv:1711.02782*, 2017.

Jiecao Yu, Jongsoo Park, and Maxim Naumov. Spatial-winograd pruning enabling sparse winograd convolution. *arXiv preprint arXiv:1901.02132*, 2019.