Approximated Oracle Filter Pruning for Destructive CNN Width Optimization

Xiaohan Ding, Guiguang Ding, Yuchen Guo, Jungong Han, Chenggang Yan

Tsinghua University, Beijing, China
University of Warwick, Coventry, UK
Hangzhou Dianzi University, Hangzhou, China

Contact: dxh17@mails.tsinghua.edu.cn

Keywords: CNN, model compression and acceleration, network pruning, filter pruning, channel pruning

Approximated Oracle Filter Pruning for Destructive CNN Width Optimization

- Filter pruning aims to remove some filters in CNNs to reduce the parameters, FLOPs, memory footprint, power consumption, etc.

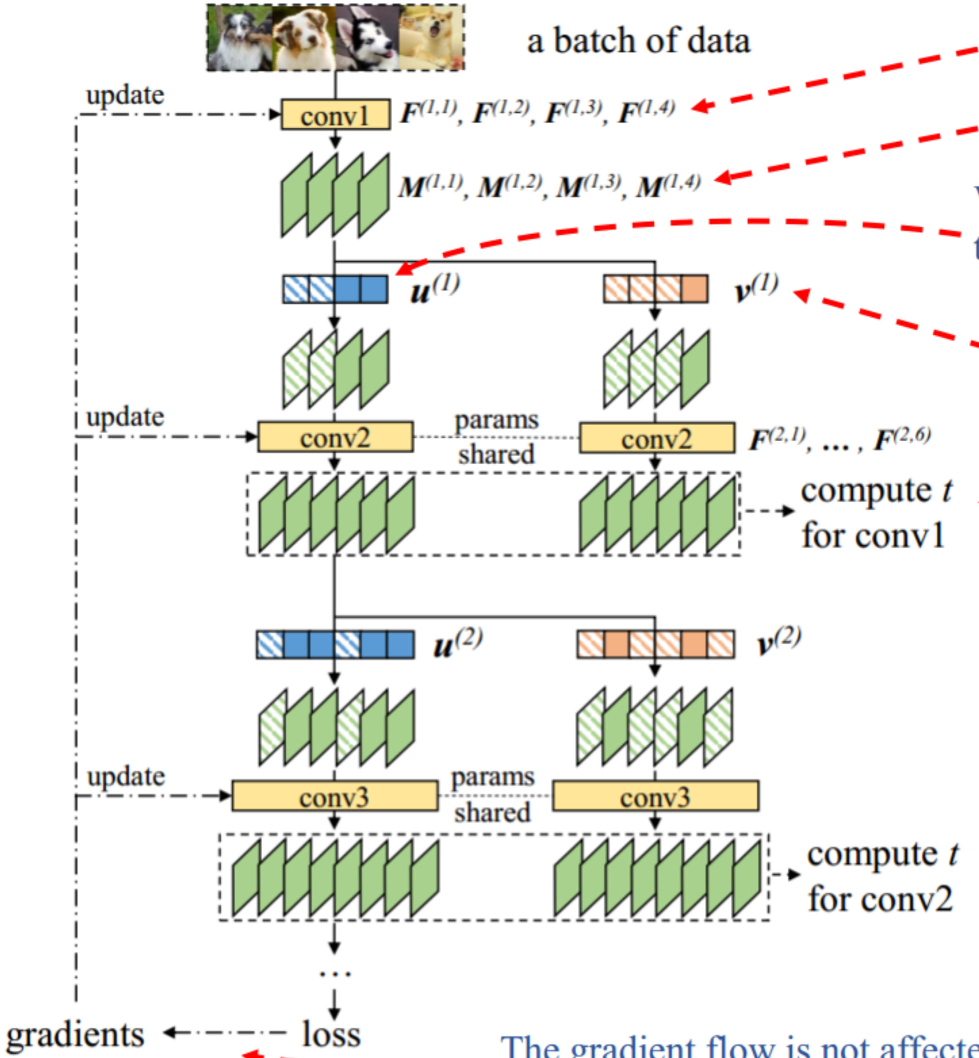
- **The problems:**
  - Given a well-trained model, it is difficult to recognize and remove the redundant filters.
  - Given a CNN architecture, it is tricky to decide the number of filters (i.e., the width) at each conv layer.

- **Our method can:**
  - shrink a wide well-trained redundant CNN into a narrower compact one (filter pruning)
  - optimize the width of each conv layer in a specific architecture (CNN Re-design)

- AOFP is a **multi-path training-time filter pruning framework**, where we keep searching for the next filters to prune in a **binary search** manner and **finetuning the model in the meantime**, which features high quality of importance estimation, reasonable time complexity and no need for heuristic knowledge

- We ablate the filters randomly, then compute and accumulate **the change in the next layer's outputs**

- **Binary Filter Search enables to automatically decide the optimal pruning granularity and eventual width of conv layers**.



4 filters

4 feature map channels

We remove a filter permanently by setting the bit in **base mask** to 0

We make temporary filter ablation attempts by setting some bits in the **scoring mask** (which is randomly altered) to 0

We use Euclidean distance (denoted by t) to measure how much the outputs of conv2 are deviated by the ablation attempt on conv1 (the 3rd filter, as shown here)

The gradient flow is not affected by the filter ablation attempts

Figure 1. Overview of AOFP, where conv1 and conv2 in a CNN are being pruned simultaneously for example. Filters $F^{(1,1)}, F^{(1,2)}, F^{(2,1)}, F^{(2,4)}$ have already been masked out, and the algorithm is trying to pick the next unimportant one out of $\{F^{(1,3)}, F^{(1,4)}\}$ and two out of $\{F^{(2,2)}, F^{(2,3)}, F^{(2,5)}, F^{(2,6)}\}$.

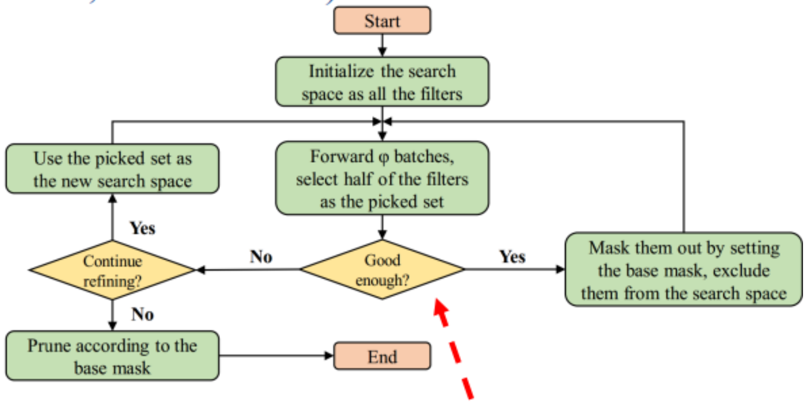

Start
Initialize the search space as all the filters
Forward φ batches, select half of the filters as the picked set
Good enough?
Yes
Mask them out by setting the base mask, exclude them from the search space
No
Continue refining?
Yes
Use the picked set as the new search space
No
Prune according to the base mask
End

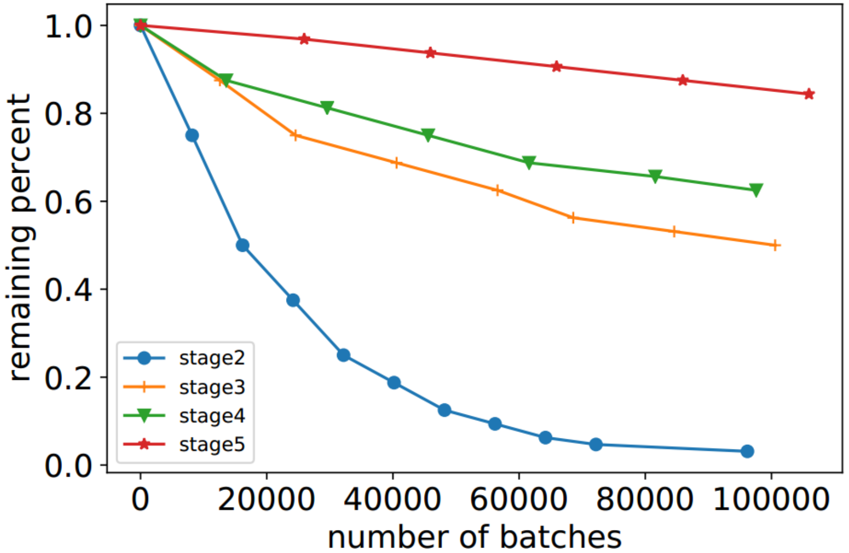
Figure 2. Flow chart of AOFP on a single layer.

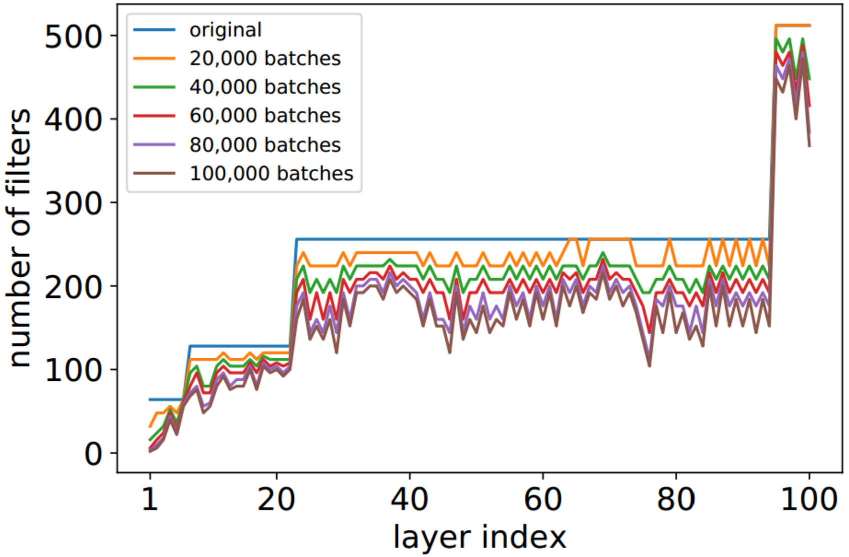
We accumulate the collected t values to discover one half of the filters which are the less important. We use a threshold to decide if the current half are good (unimportant) enough. If not, we continue to find the less important half from them (i.e., 1/4 of the original search space).

Approximated Oracle Filter Pruning for Destructive CNN Width Optimization

•Pruning an existing model:

- As AOFP proceeds on ResNet-152, we show the remaining percentage of filters at the first layers in the four stages as the representatives (left, which originally have 64, 128, 256 and 512 filters), and remaining width of all the target layers (right) every 20,000 batches. As can be observed, AOFP automatically figures out that the first layer in stage2 can be pruned significantly, and chooses to prune it with large granularity (8 filters every time) at the beginning, then gradually reduces the granularity.

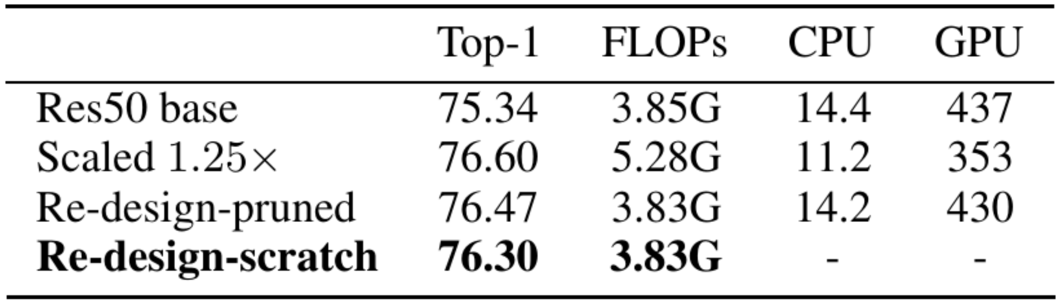




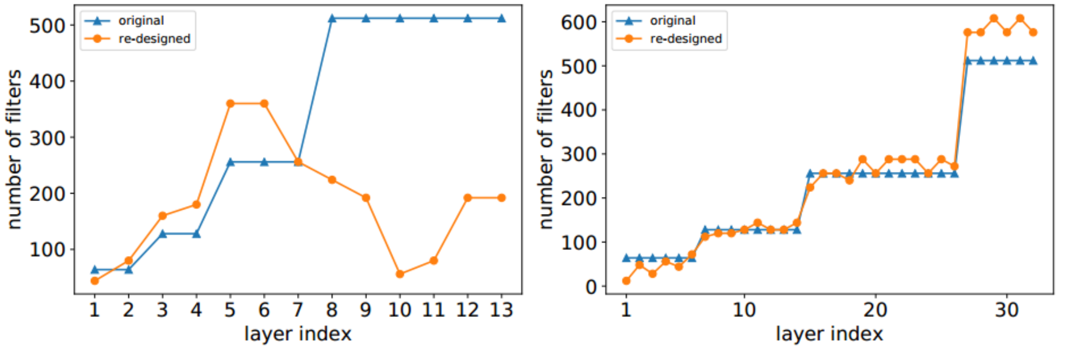
Approximated Oracle Filter Pruning for Destructive CNN Width Optimization

•CNN Re-design:

- We train a scaled ResNet-50 where the 1st and 2nd layers in each residual block have 1.25X of the original width, then use AOFP to reduce its FLOPs to the same level as the original ResNet-50. In this way, **we obtain a network where some layers are wider than the original ResNet-50 and some are narrower**. We train a model with the discovered structure from scratch, and the accuracy is still higher than the baseline. It is observed that the irregularly shaped structure runs as fast as the tidy baseline (measured in examples/sec).



	Top-1	FLOPs	CPU	GPU
Res50 base	75.34	3.85G	14.4	437
Scaled 1.25×	76.60	5.28G	11.2	353
Re-design-pruned	76.47	3.83G	14.2	430
Re-design-scratch	**76.30**	**3.83G**	-	-



(a) VGG on CIFAR10. (b) ResNet-50 on ImageNet.

Figure 7. Layer width of the re-designed models in comparison with the original. Note again that only the internal layers of ResNet-50 (i.e., the first two layers in each residual block) are shown.

Approximated Oracle Filter Pruning for Destructive CNN Width Optimization

- Thank you for your attention!
- Welcome to our poster:
  - Wed Jun 12th 06:30 -- 09:00 PM
  - Room: Pacific Ballroom