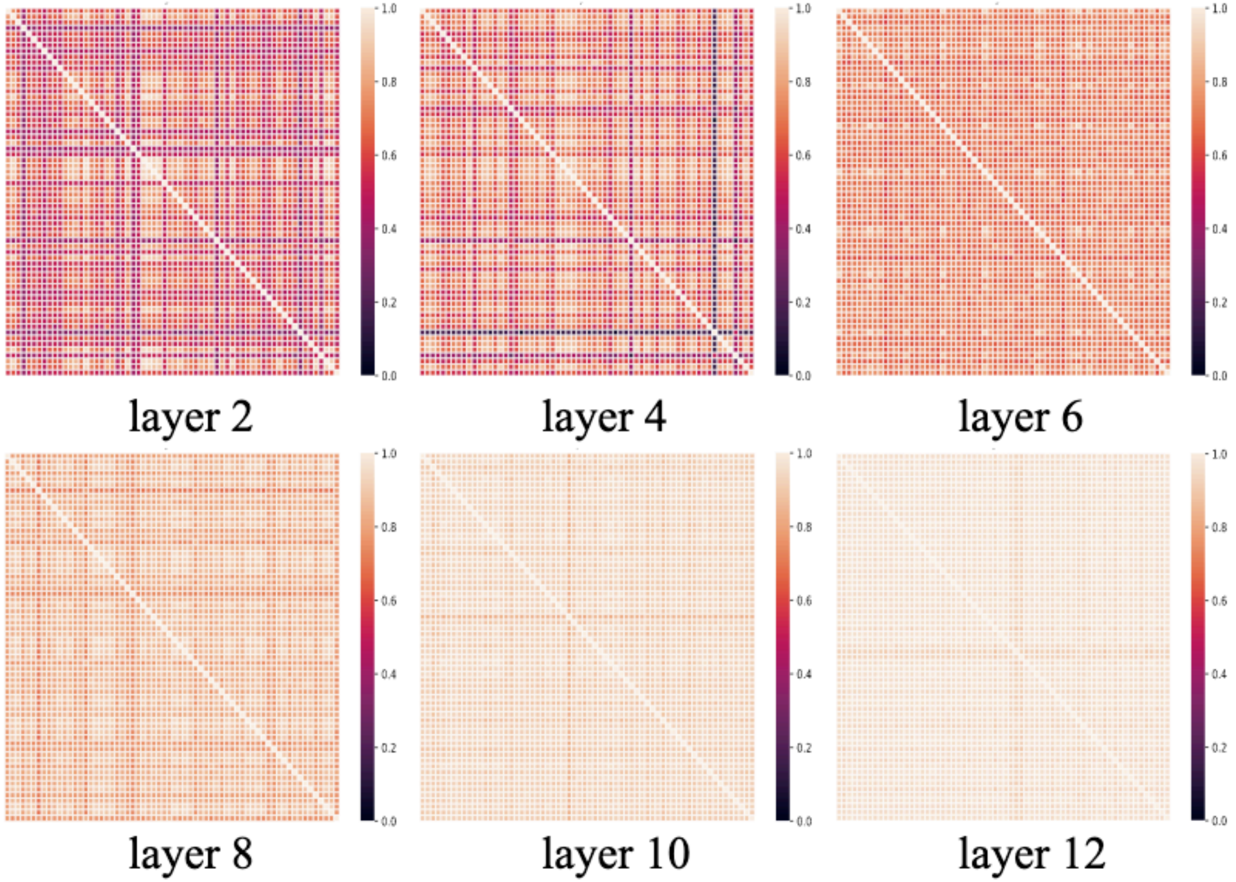
Tied Block Convolution: Leaner and Better CNNs with Shared Thinner Filters

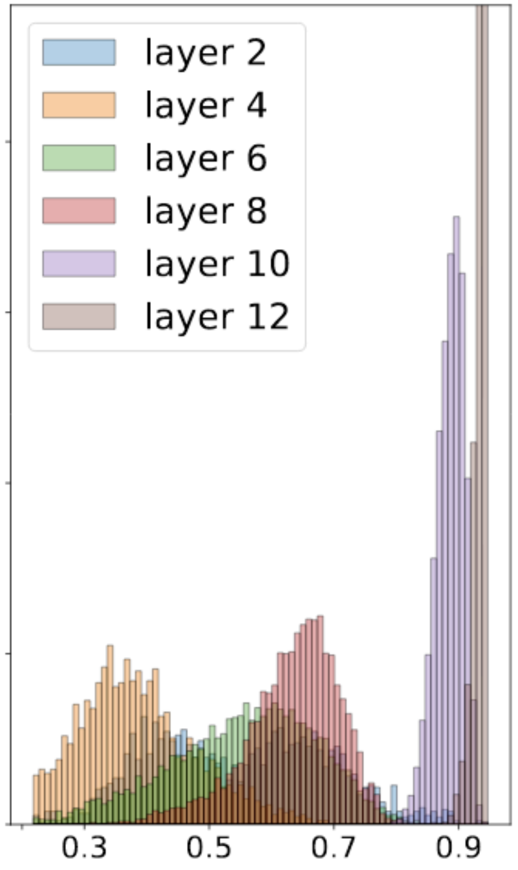
Xudong Wang[1,2] and Stella X. Yu[1,2]
[1]University of California, Berkeley; [2]International Computer Science Institute

Presented by:
Xudong (Frank) Wang
Ph.D. Student @ EECS at UC Berkeley
Graduate Student Researcher @ International Computer Science Institute

Motivation

Filters of an optimized CNN become more similar at an increasing depth.



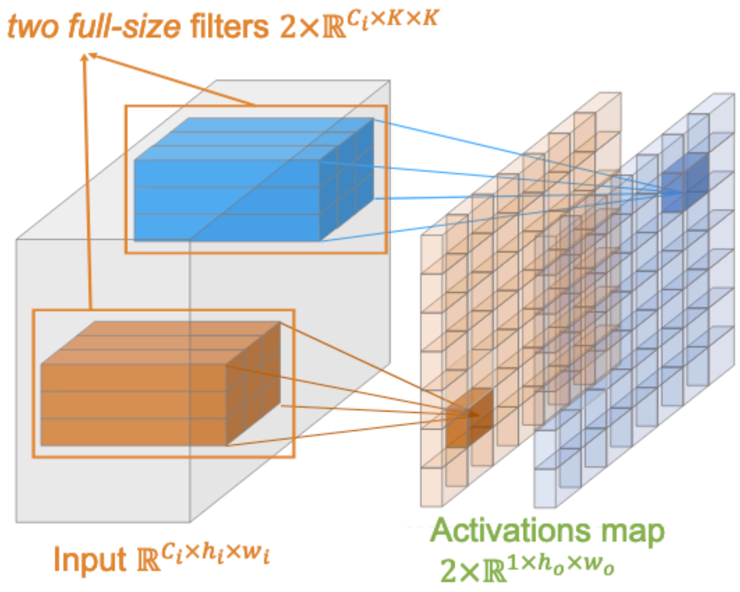


Correlation matrix of 64 randomly selected filters based on their guided back-propagation patterns [1].

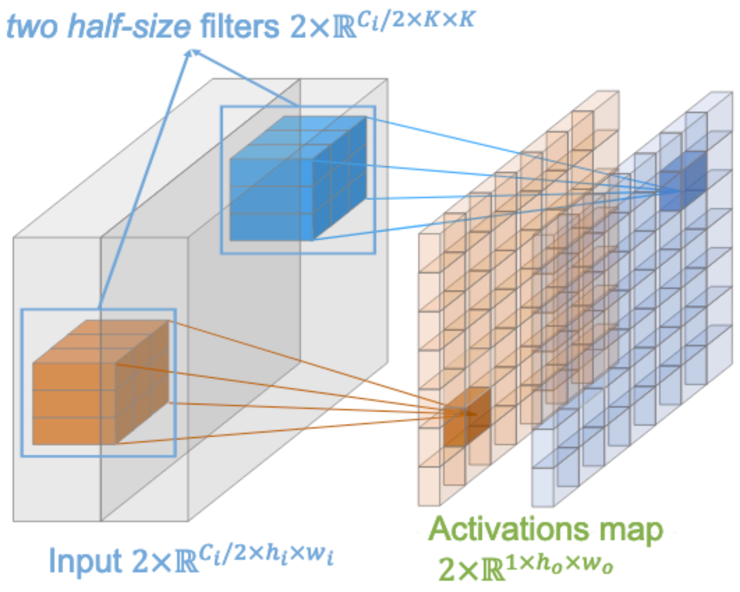
Normalized histograms of pairwise filter similarities

[1] Springenberg, J.T., et al., 2014. Striving for simplicity: The all convolutional net. arXiv preprint arXiv:1412.6806.

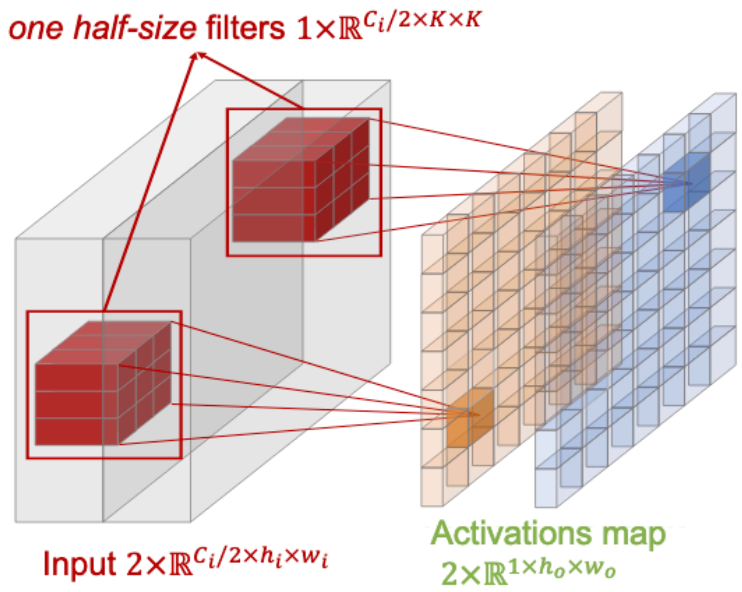
Tied Block Convolution (TBC)



Standard Convolution (SC)



Group Convolution (GC)



Tied Block Convolution (TBC)

Tied Block Convolution (TBC) vs Group Convolution (GC)

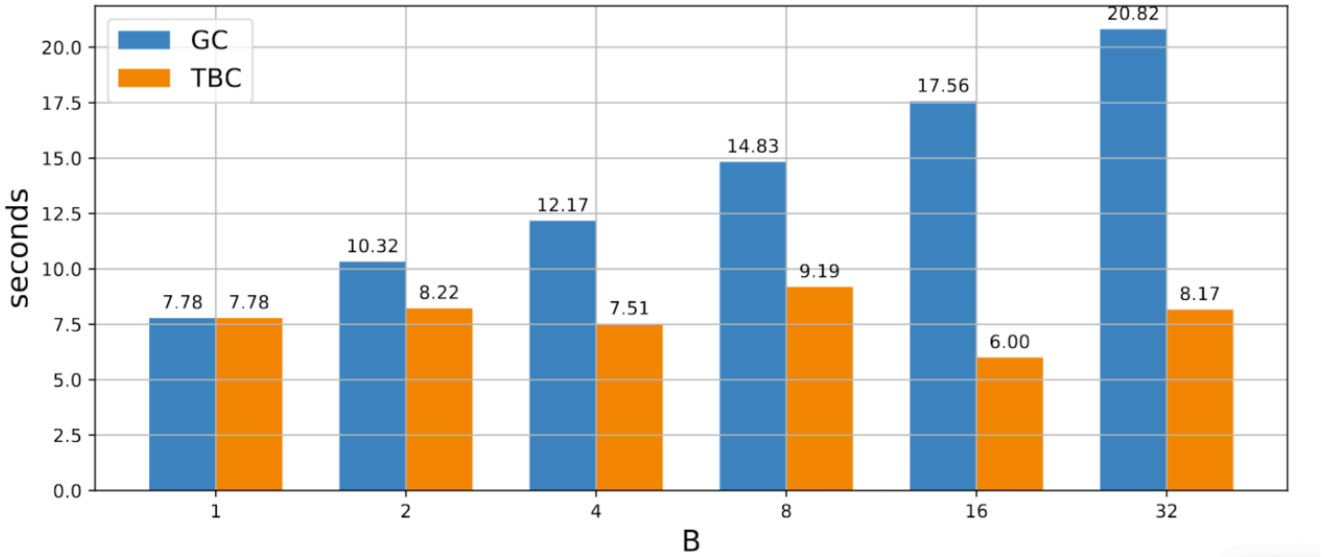
TBC has several major distinctions from GC in practical consequences (assume that the block number B of TBC is the same as the group number G of GC)

- TBC has B× fewer parameters than GC.

Tied Block Convolution (TBC) vs Group Convolution (GC)

TBC has several major distinctions from GC in practical consequences (assume that the block number B of TBC is the same as the group number G of GC)

- TBC has B× fewer parameters than GC.
- TBC only has one fragmentation on GPU utilization, where as GC has G fragmentations, greatly reducing the degree of parallelism.



The time cost of processing 1k iterations of each feature map using the RTX 2080Ti GPU. Input feature map size is 56 × 56 × 2048.

Tied Block Convolution (TBC) vs Group Convolution (GC)

TBC has several major distinctions from GC in practical consequences (assume that the block number B of TBC is the same as the group number G of GC)

- TBC has B× fewer parameters than GC.
- TBC only has one fragmentation on GPU utilization, where as GC has G fragmentations, greatly reducing the degree of parallelism.
- TBC can better model cross-channel dependencies.

Tied Block Convolution (TBC) vs Group Convolution (GC)

TBC has several major distinctions from GC in practical consequences (assume that the block number B of TBC is the same as the group number G of GC)

- TBC has B× fewer parameters than GC.
- TBC only has one fragmentation on GPU utilization, where as GC has G fragmentations, greatly reducing the degree of parallelism.
- TBC can better model cross-channel dependencies.
- TBC-based TiedResNet greatly surpasses GC-integrated ResNeXt on object detection and instance segmentation tasks.

Tied Block Group Convolution (TGC)

Group Convolution (GC)

$$\tilde{X} = X_1 * W_1 \oplus X_2 * W_2 \oplus \cdots \oplus X_G * WG$$

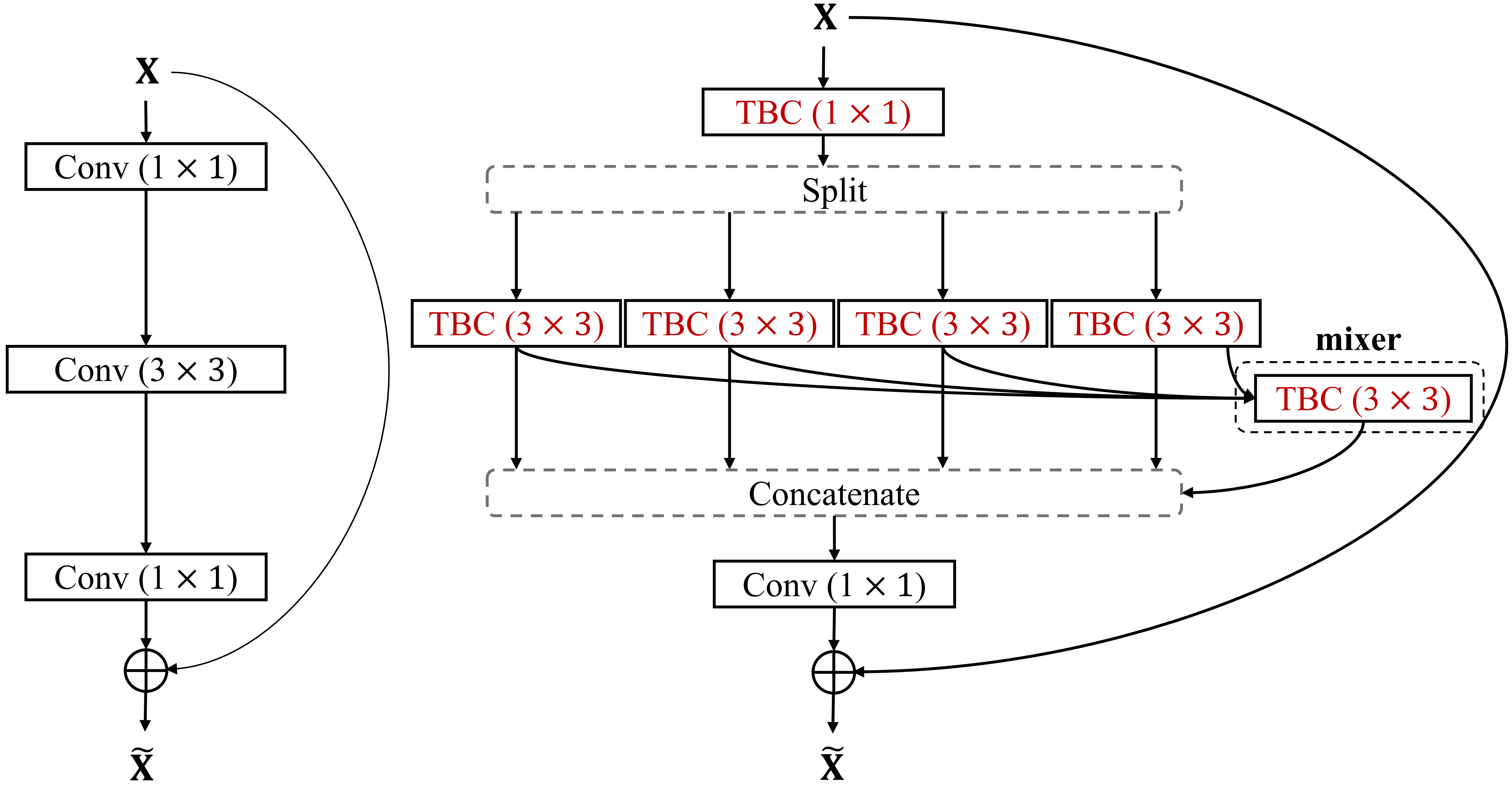
Where $\oplus$ is the concatenation operation along the channel dimension, W_g is the convolution filters for group g, where $g \in \{1, \dots, G\}$, $X_g \in R^{\frac{C_i}{G} \times h_i \times w_i}$, $W_g \in R^{\frac{C_o}{G} \times \frac{C_i}{G} \times k \times k}$

Tied Block Group Convolution (TGC)

$$\tilde{X} = (X_{11} * W_1' \oplus \cdots \oplus X_{1B} * W_1') \oplus \cdots \oplus (X_{G1} {}_* W_G' \oplus \cdots \oplus X_{GB} * W_G')$$

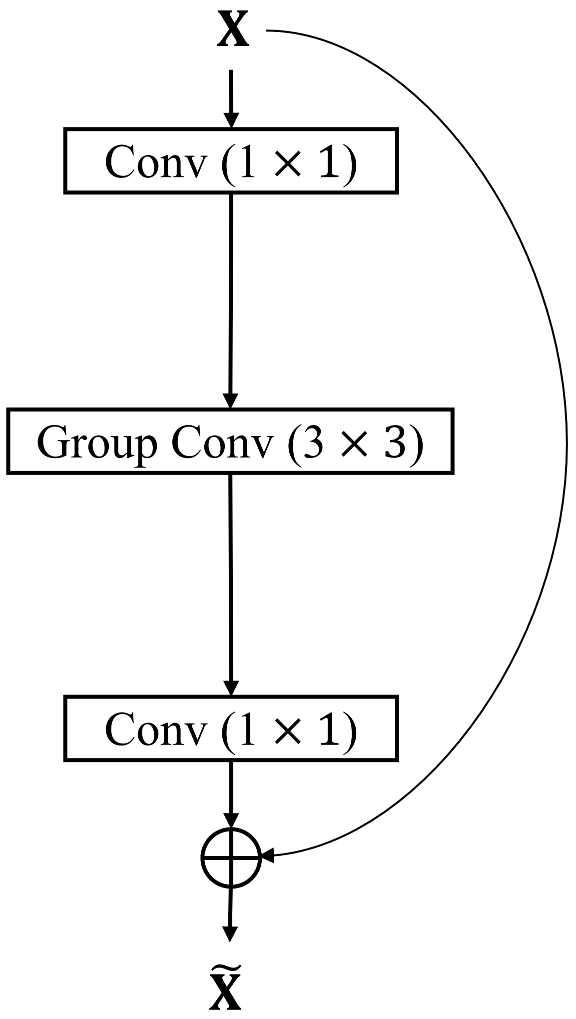
Where $g \in \{1, \dots, G\}, b \in \{1, \dots, B\}, X_{gb} \in R^{\frac{C_i}{BG} \times h_i \times w_i}, W_g' \in R^{\frac{C_o}{BG} \times \frac{C_i}{BG} \times k \times k}$

TBC in ResNet

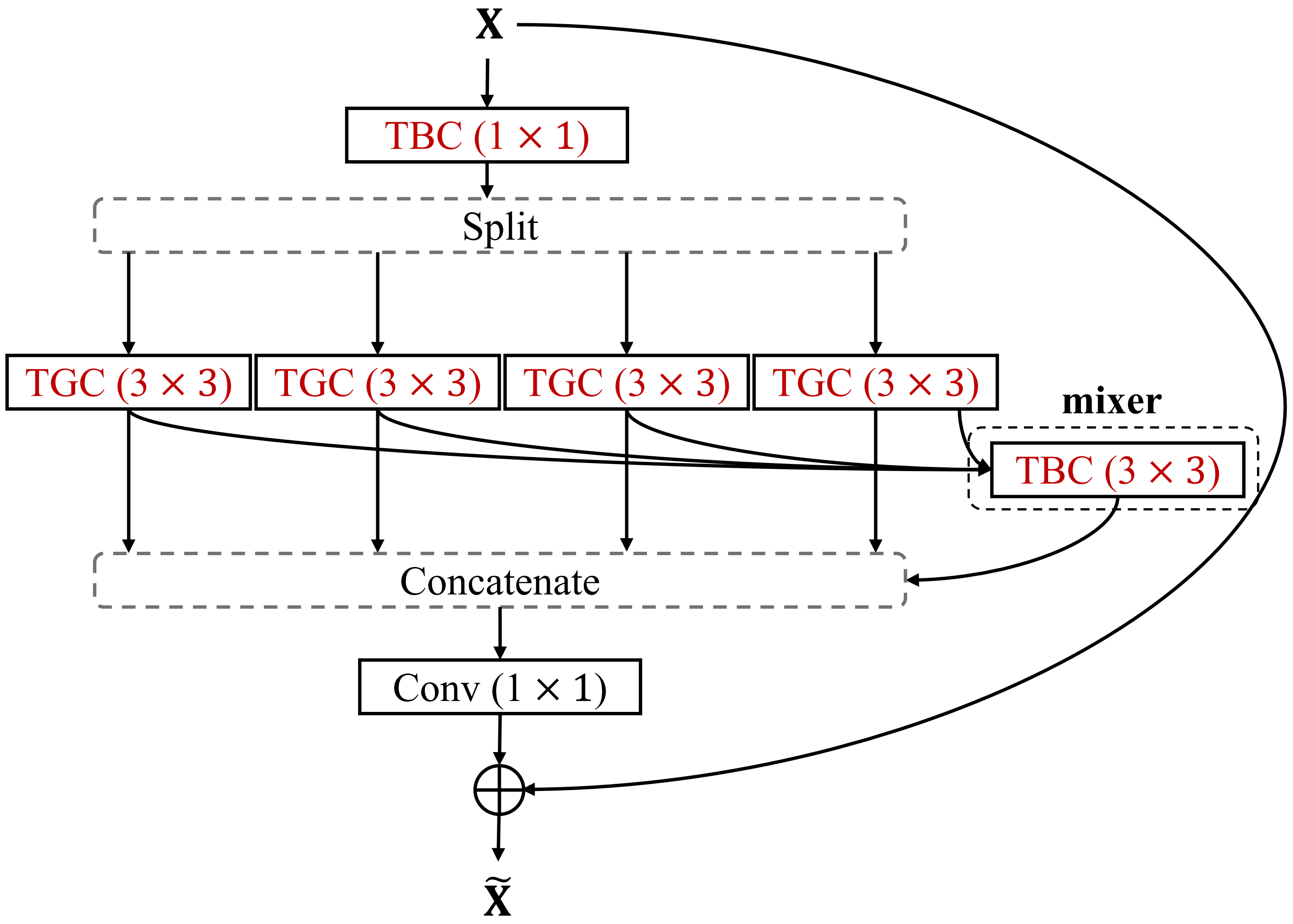


ResNet Bottleneck **TiedResNet Bottleneck**

TGC/TBC in ResNeXt

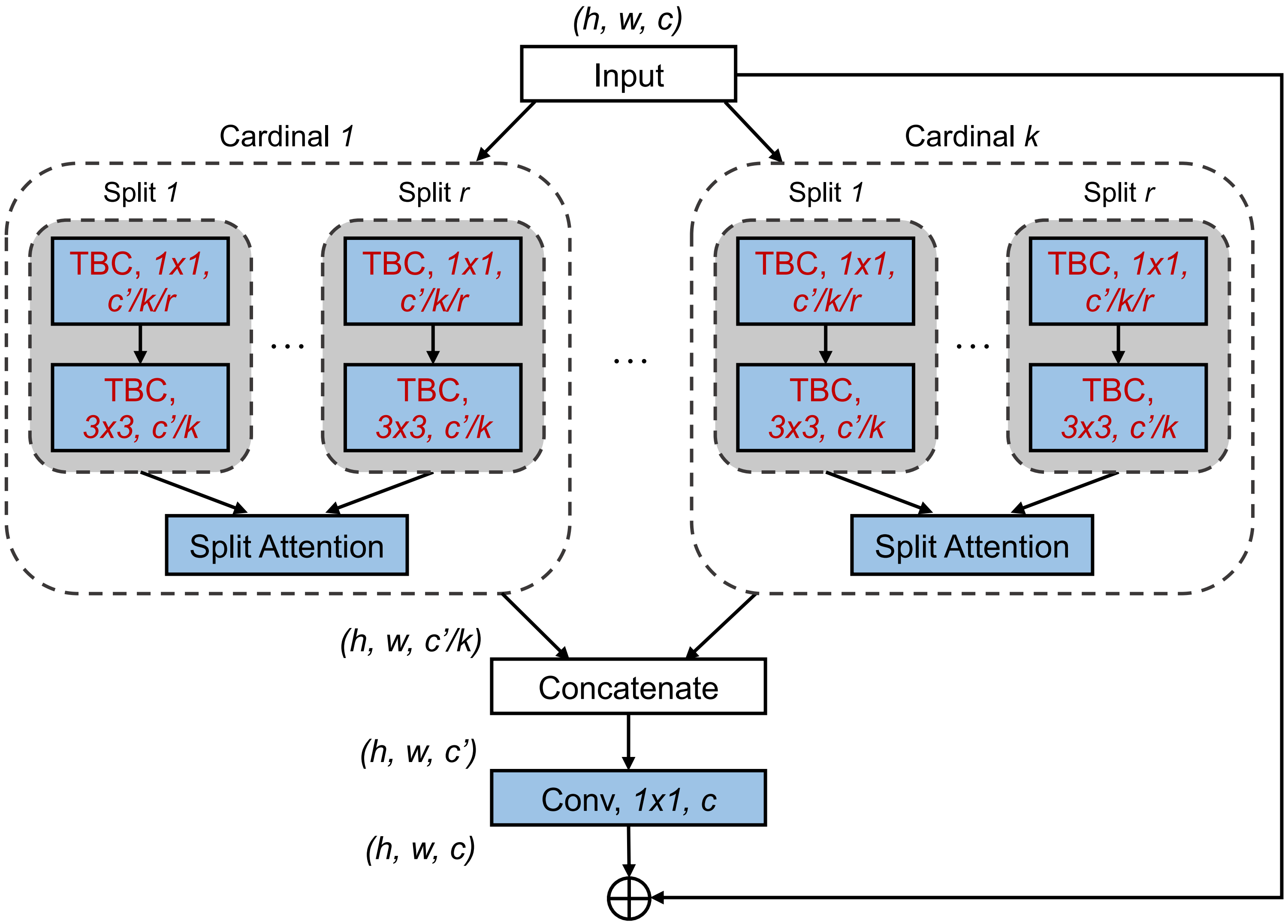


ResNeXt Bottleneck



TiedResNeXt Bottleneck

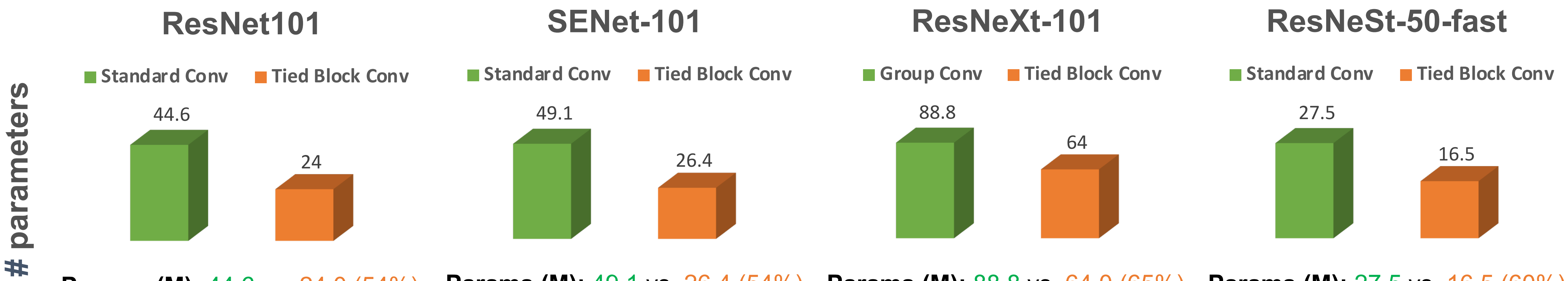
TBC in ResNeSt



TiedResNeSt Bottleneck

Recognition (ImageNet)

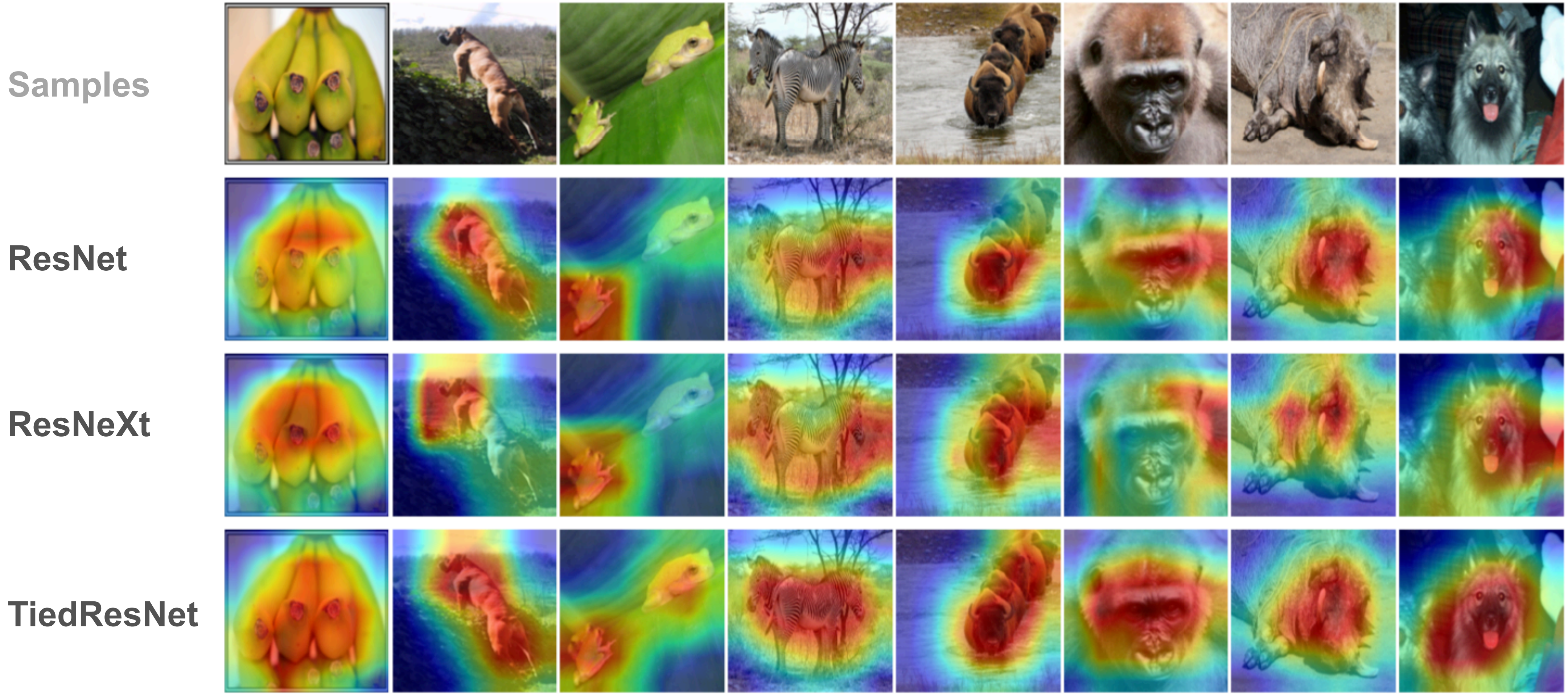
The integration of TBC/TFC/TGC can obtain consistent performance improvements to various backbone networks.



ResNet101	SENet-101	ResNeXt-101	ResNeSt-50-fast
Params (M): 44.6 vs. 24.0 (54%)	**Params (M):** 49.1 vs. 26.4 (54%)	**Params (M):** 88.8 vs. 64.0 (65%)	**Params (M):** 27.5 vs. 16.5 (60%)
Top-1 acc (%): 77.4 vs. 77.7	**Top-1 acc (%):** 78.3 vs. 79.0	**Top-1 acc (%):** 79.3 vs. 79.3	**Top-1 acc (%):** 78.6 vs. 78.8

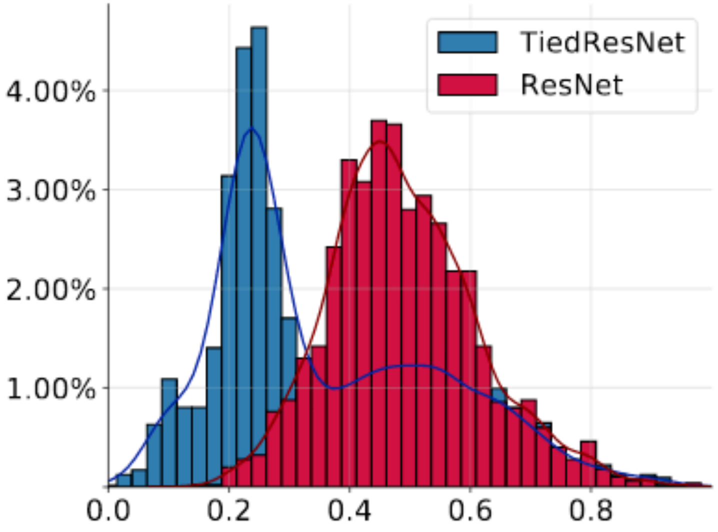
Grad-CAM Visualization

TiedResNet focusing on target objects more properly than ResNet and ResNeXt.

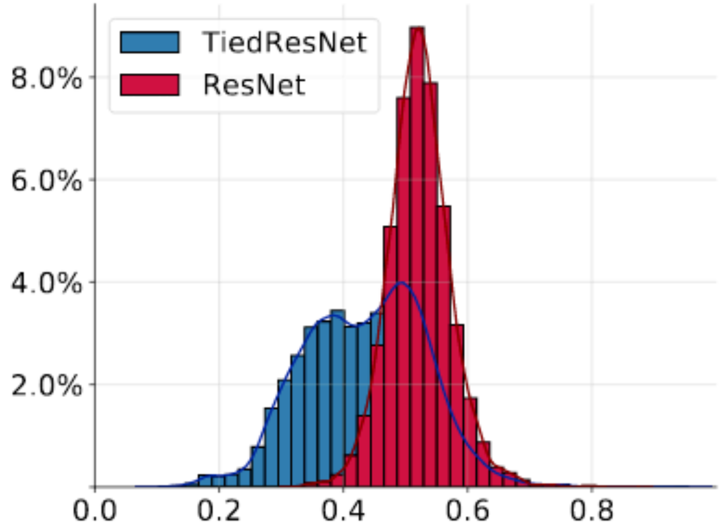


Filter Similarity

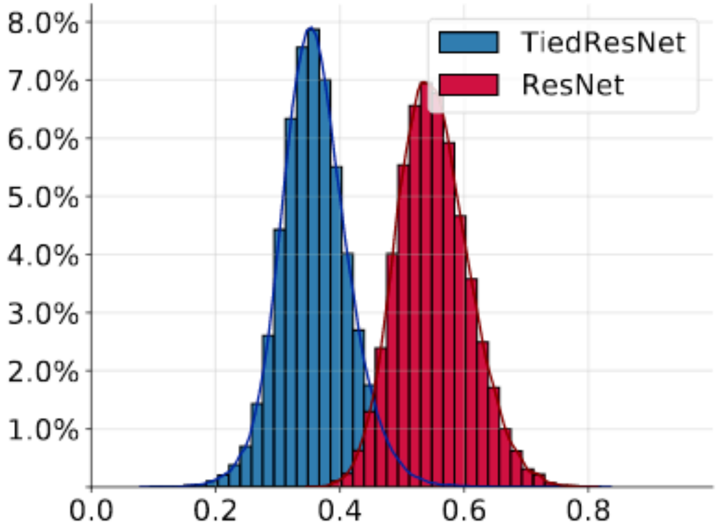
TiedResNet learns less correlated filters than ResNet.



Layer 9



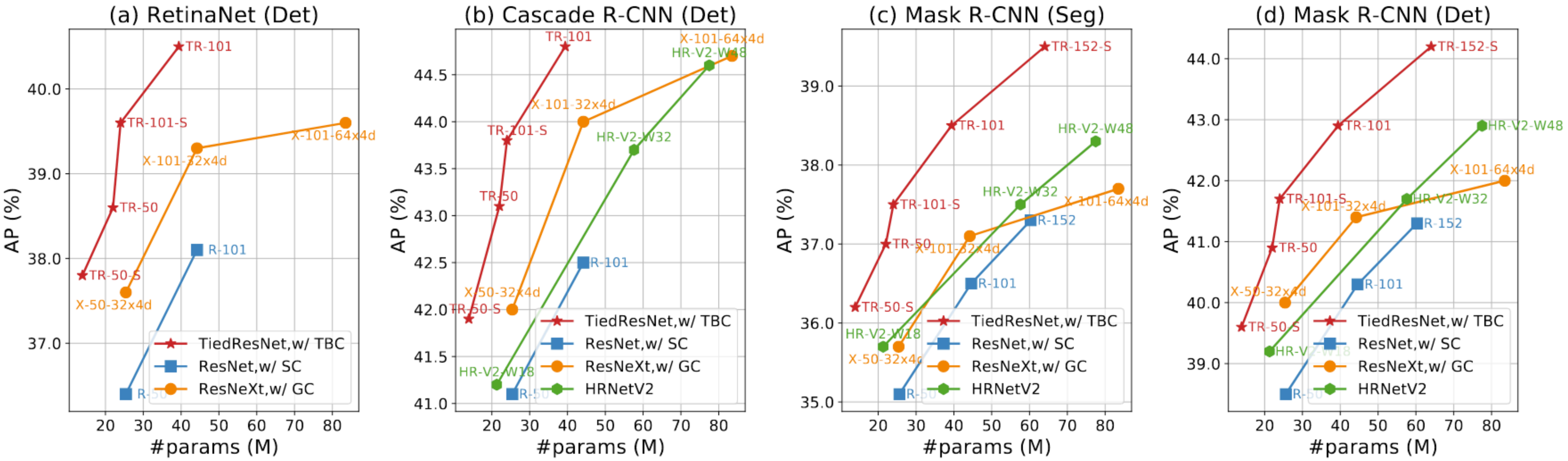
Layer 18



Layer 36

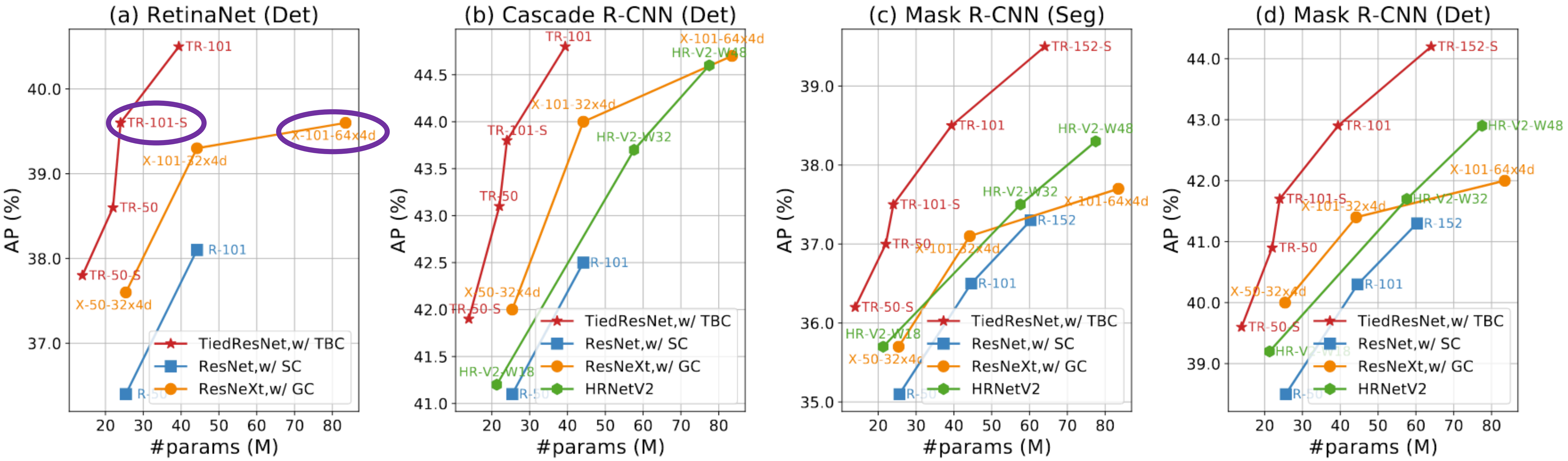
Detection and Segmentation (MS-COCO)

TiedResNet consistently outperforms ResNet, ResNeXt and HRNetV2 with much fewer parameters.



Detection and Segmentation (MS-COCO)

TiedResNet consistently outperforms ResNet, ResNeXt and HRNetV2 with much fewer parameters.



Object Detection on Pascal VOC

With only 31% parameters,TiedResNet50-S reaches comparable performance with ResNet101.

Framework	Backbone	Params (M)	mAP (%)
SSD513 [27]	VGG16 [33]	138.36	80.6
RefineDet512 [44]	VGG16 [33]	138.36	81.8
R-FCN [10]	ResNet101 [17]	44.65	80.5
DSSD513 [13]	ResNet101 [17]	44.65	81.5
CoupleNet [46]	ResNet101 [17]	44.65	82.7
Faster R-CNN with FPN [24]	ResNet50 [17]	25.56	80.9
Faster R-CNN with FPN [24]	ResNet101 [17]	44.65	82.1
Faster R-CNN with FPN [24]	TiedResNet50-S	**13.91**	81.9
Faster R-CNN with FPN [24]	TiedResNet50	22.03	82.6
Faster R-CNN with FPN [24]	TiedResNet101-S	23.98	82.9
Faster R-CNN with FPN [24]	TiedResNet101	39.43	**83.8**

Instance Segmentation on Cityscapes

TiedResNet50 can reach 2.1% gain for AP^{mask}

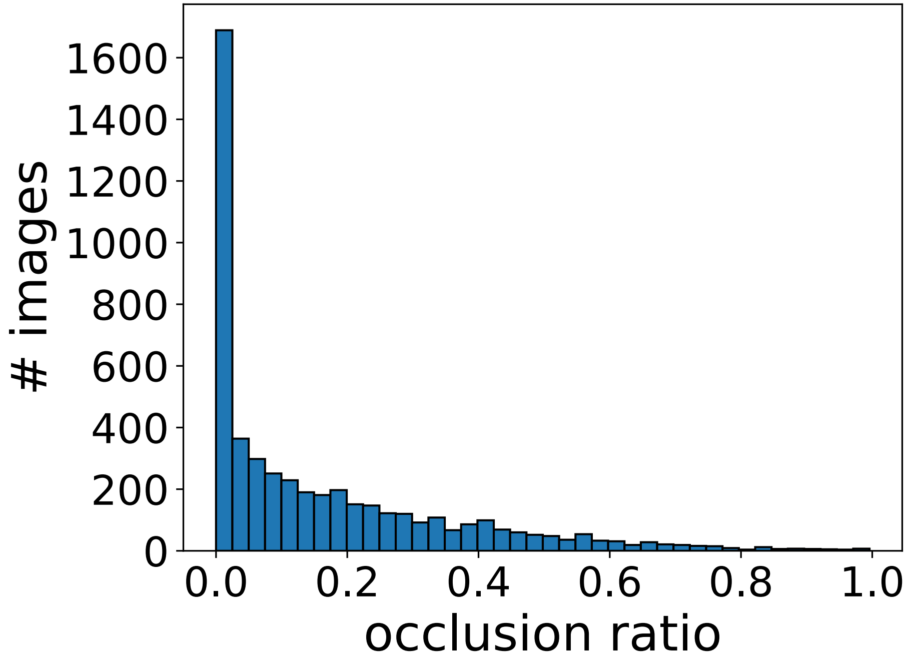
Framework	Backbone	Params (M)	AP^{mask}
Mask R-CNN [16]	ResNet50 [17]	25.6	31.5
Mask R-CNN [16]	TiedResNet50-S	13.9	32.5
Mask R-CNN [16]	TiedResNet50	22.0	**33.6**

Object Detection Under High Occlusion Ratios

The occlusion ratio (r) of each image is evaluated by:

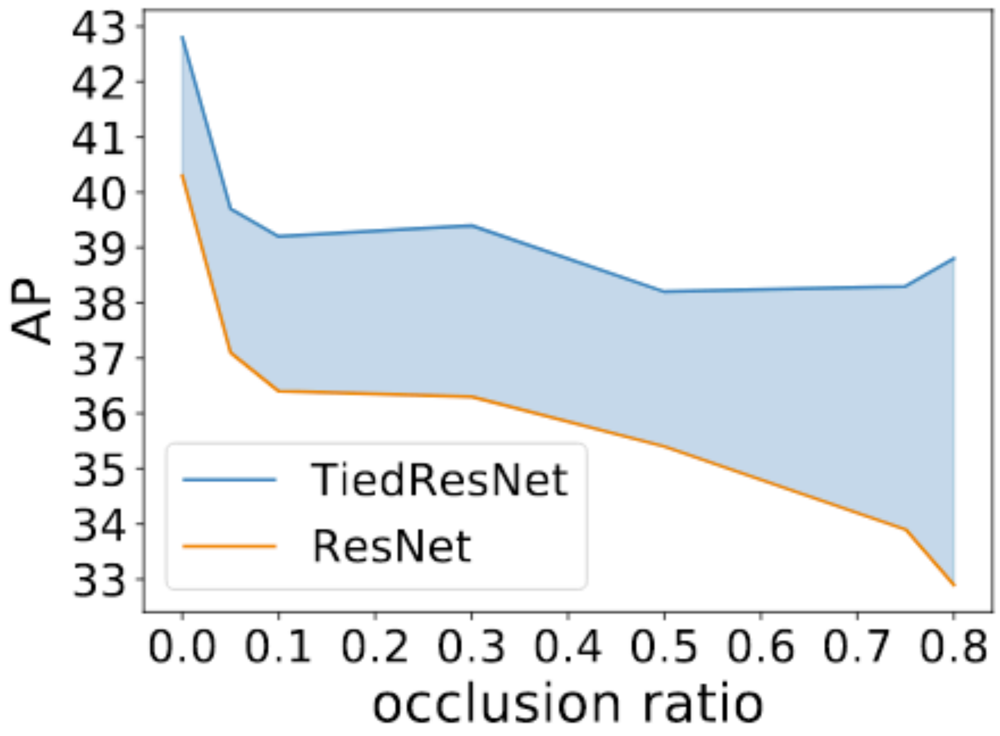
$$r = \frac{\text{total overlap area}}{\text{total instance area}}$$

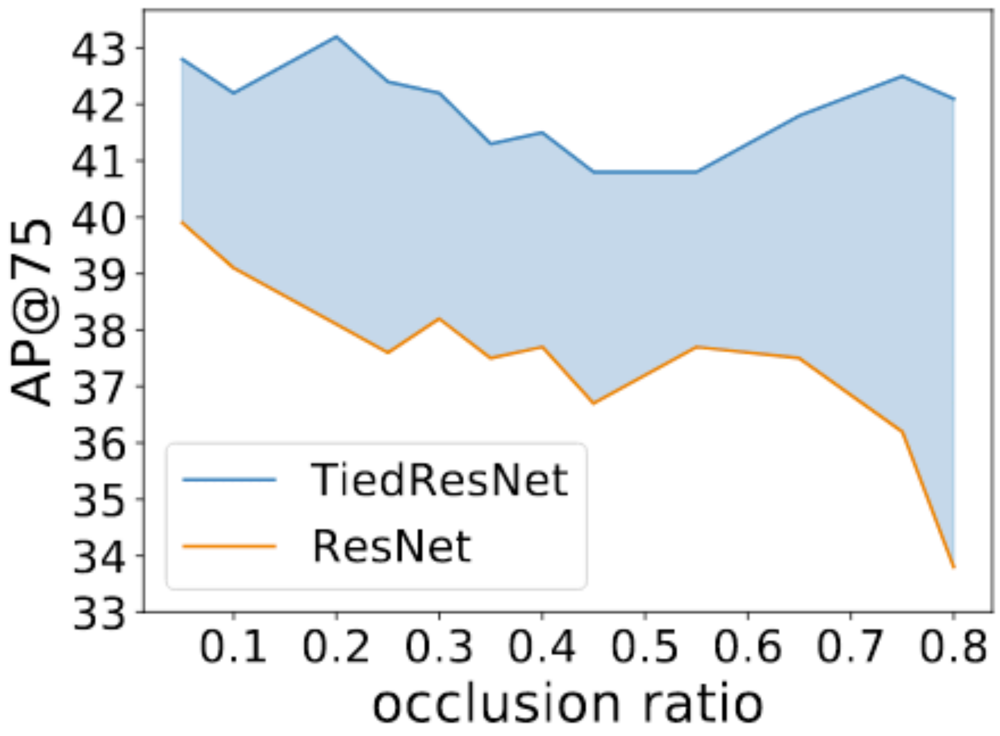
The number of images relative to the instance occlusion ratio r in MS-COCO val-2017split.



Object Detection Under High Occlusion Ratios

When r= 0.8, TiedResNet increases by 8.3% at AP^{75} and 5.9% at AP, much more effective at handling highly overlapping instances.



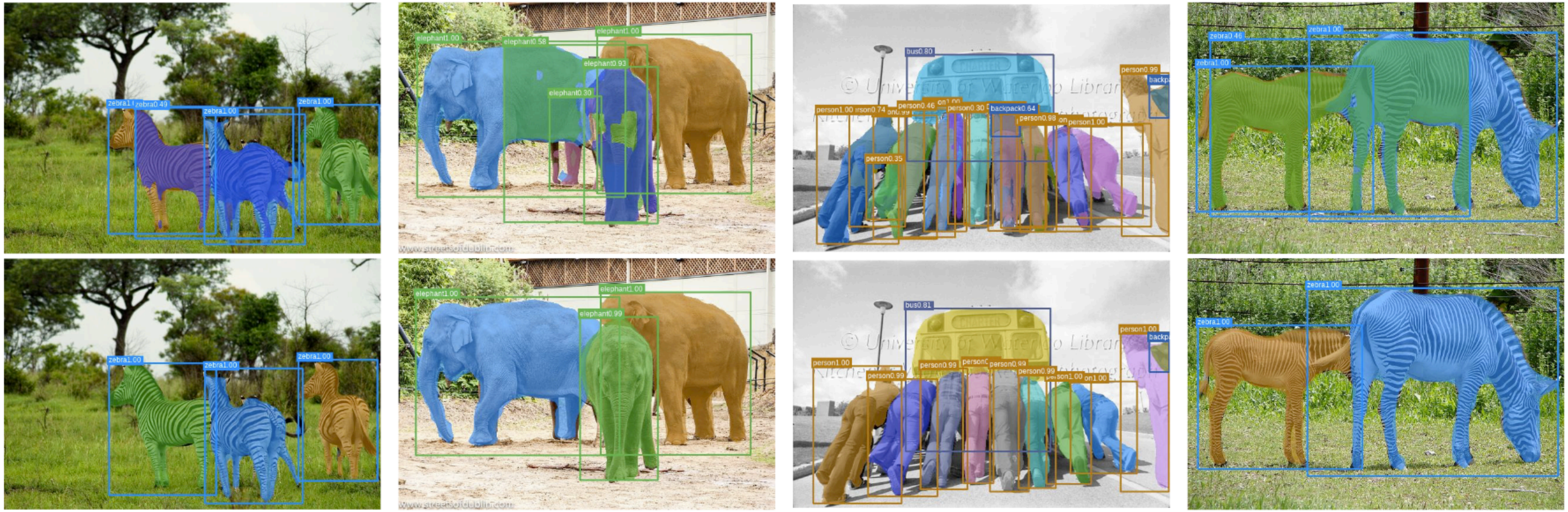


Object Detection Under High Occlusion Ratios

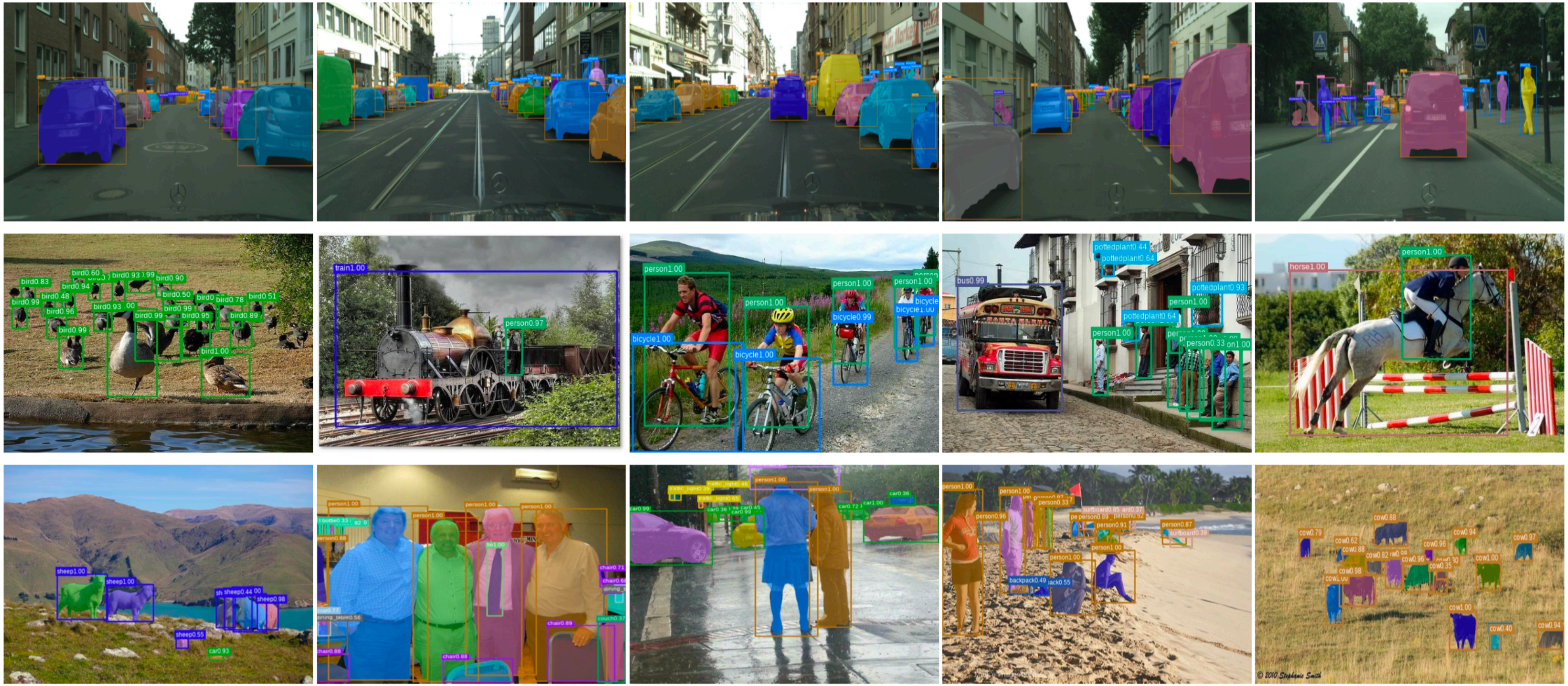
Fewer false positive and false negative proposals

ResNet

TiedResNet

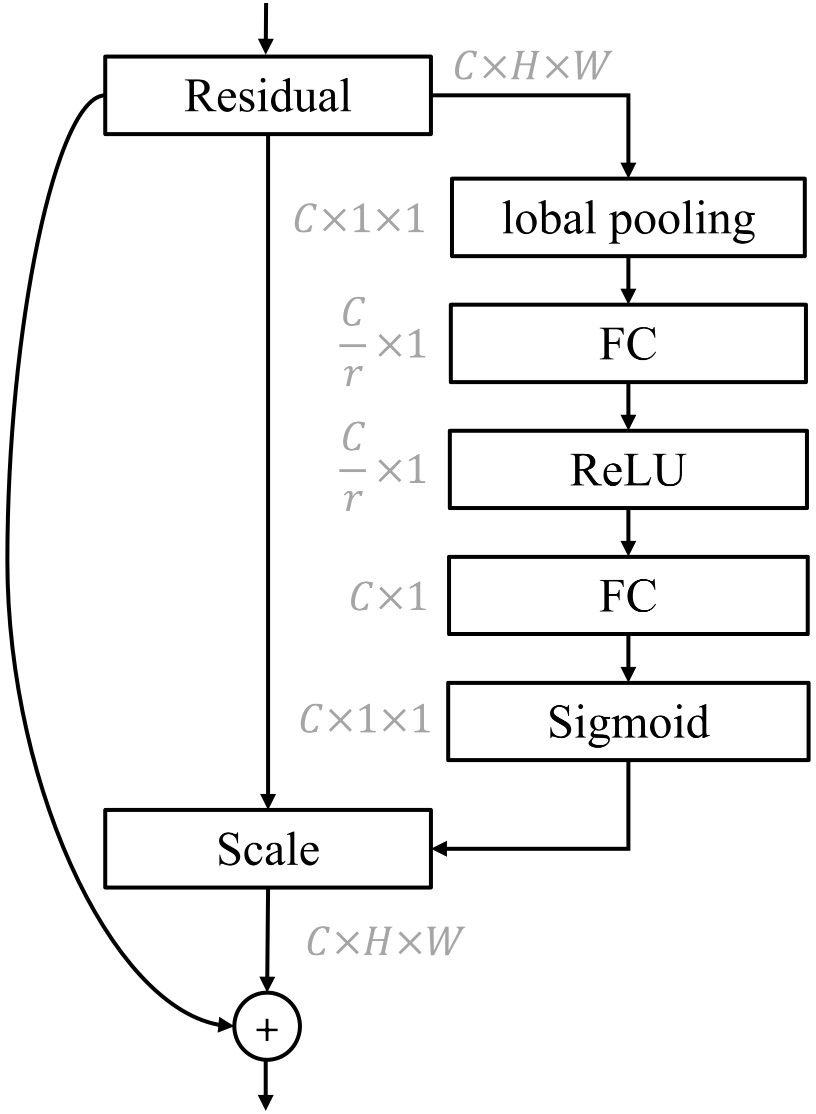


Sample Results (Cityscapes, Pascal VOC, MS-COCO)

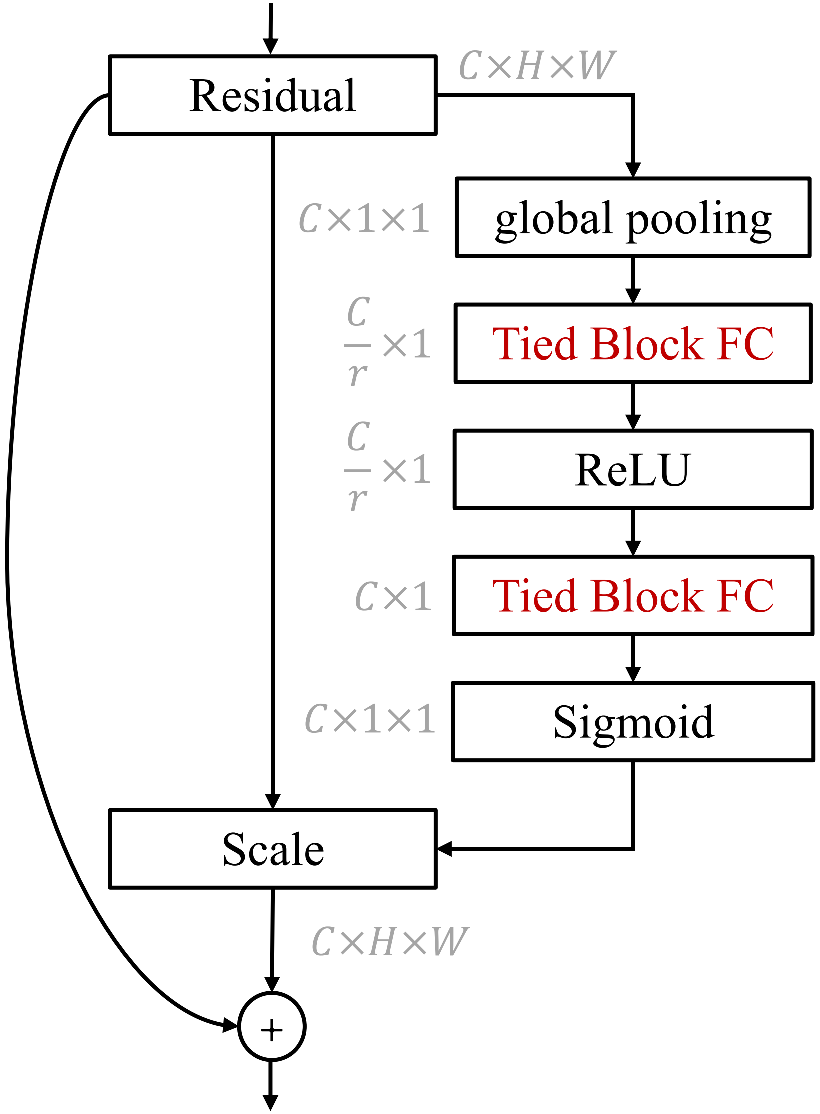


Attention Modules: SE and TiedSE

SE



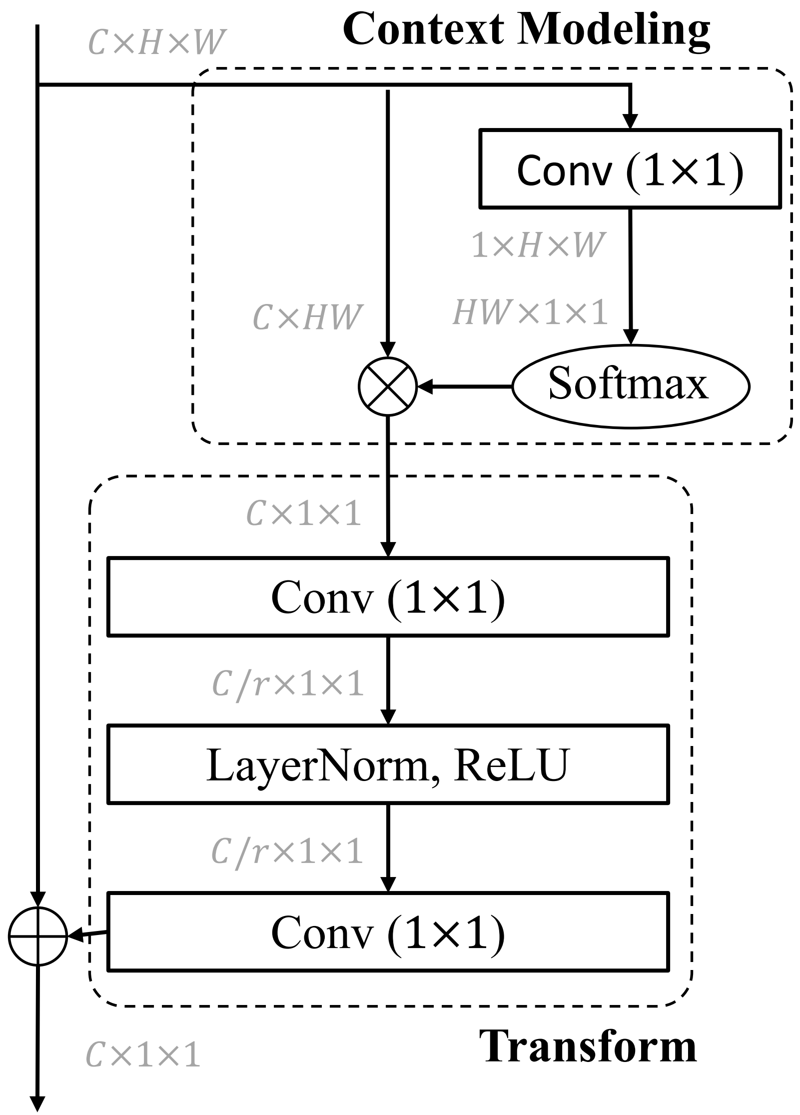
TiedSE



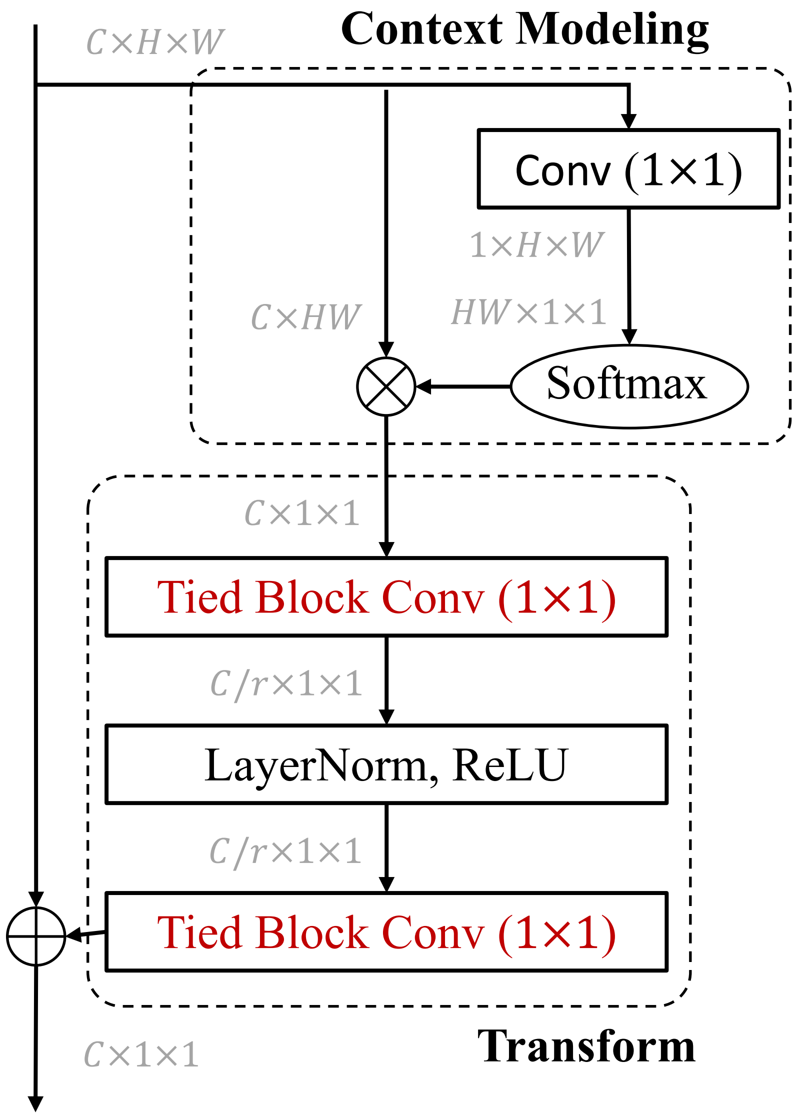
Hu, Jie, Li Shen, and Gang Sun. "Squeeze-and-excitation networks." CVPR. 2018.

Attention Modules: Global Context (GC) and TiedGC

GC



TiedGC



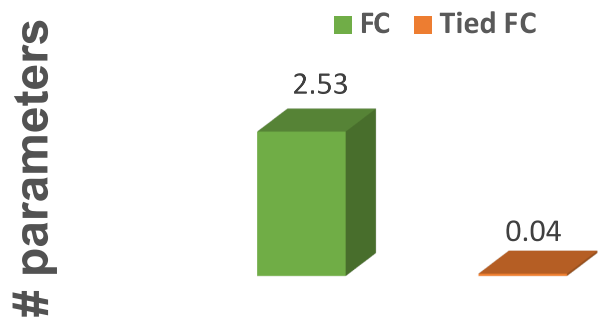
Cao, Y., Xu, J., Lin, S., Wei, F. and Hu, H.. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. ICCVW 2019.

Attention Module (TiedSE and TiedGC)

Significantly reduce attention module parameters with comparable performance

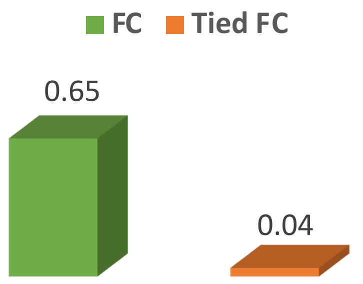
ImageNet

SE + ResNet50



Params (M): 2.53 vs. 0.04 (1.6%)

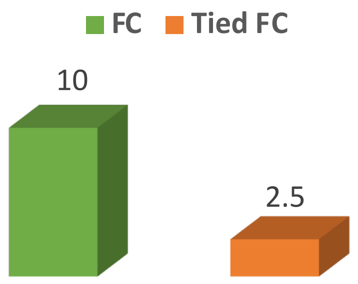
SE + EfficientNet-B0



Params (M): 0.65 vs. 0.04 (6.4%)

MS-COCO

Global Context + Mask-RCNN



Params (M): 10.0 vs. 2.5 (25%)

Summary

- **The proposed Tied Block Convolution (TBC) reduce $B^2\times$ parameters and $B\times$ computational cost;**
- **The concept of TBC can be extended to group convolution and fully connected layers;**
- **TBC/TGC/TFC can be applied to various backbone networks and attention modules;**
- **Our extensive experimentation on classification, detection, instance segmentation, and attention demonstrates TBC's significant across-the-board gain over standard convolution and group convolution;**

Tied Block Convolution: Leaner and Better CNNs with Shared Thinner Filters

Xudong Wang [1,2] and Stella X. Yu [1,2]
[1] University of California, Berkeley
[2] International Computer Science Institute
xdwang@eecs.berkeley.edu, stellayu@berkeley.edu

Project Page **Code** **Preprint**