A. Implementation Details

A.1. Image Classification

In accordance with Stand-Alone Self-Attention [13] and Axial Attention [15], we train all these models for 130 epochs utilizing the Stochastic Gradient Descent (SGD) optimizer with the momentum of 0.9 and the weight decay of 0.0001. The learning rate initiates from 0.8 and gradually approaches zero following a half-cosine function shaped schedule. The mini-batch size per GPU is set to 32 and the training procedure is conducted on 64 GPU devices in total. The label smoothing regularization technique is applied with the coefficient of 0.1.

A.2. Object Detection and Instance Segmentation

Following the widely-adopted pipeline, the input images are resized to keep their shorter/longer side as 800/1333 pixels prior to being fed into the networks. The training procedure lasts for 12 epochs, using the Stochastic Gradient Descent (SGD) optimizer with the momentum of 0.9 and weight decay of 0.0001. The initial learning rate is set to 0.02 for Faster/Mask R-CNN and 0.01 for RetinaNet with a linear warm-up period of 500 iterations, divided by 10 in the 8^{th} and 11^{st} epoch. When necessary, we moderately extend the warm-up period and apply gradient clipping for the sake of convergence stability. The detectors are trained on 8 Tesla V100 GPUs with 2 samples per GPU.

A.3. Semantic Segmentation

The urban scene images with a high resolution of 1024×2048 are randomly resized, with their aspect ratios kept in the range from 0.5 to 2.0, from which the input image patches with the size of 512×1024 are randomly cropped, then undergo random horizontal flipping and a sequence of photometric distortions as the data augmentation. We adopt the training schedule of 80k iterations, and apply the Stochastic Gradient Descent (SGD) optimizer with the momentum of 0.9 and weight decay of 0.0005. The learning rate starts from 0.01 and anneals following the conventional "poly" policy, which indicates the initial learning rate is multiplied by $(1 - \frac{iter}{total\_iter})^{0.9}$ in each iteration. The segmentation networks are trained on 4 Tesla V100 GPUs with 2 samples per GPU. We apply synchronized Batch Normalization [10] for more stable estimation of the batch statistics.

B. Comparison to State-of-the-art on COCO

For both object detection and instance segmentation on COCO, we compare our involution-based Mask R-CNN [4] with the RedNet-50 backbone against other celebrated architectures with ResNet-50 in Table 1. Our approach performs substantially better than convolution-based Mask R-CNN equipped with self-attention blocks, like NLNet [16], CCNet [7], and GCNet [2]. Additionally, our method outperforms those of embedding dynamic mechanism into the networks, including Deformable ConvNets (DCN) [20] and Dynamic Graph Message passing Networks (DGMN) [19]. Note that all these referred approaches introduce extra parameters and FLOPs to the vanilla Mask R-CNN by appending complementary modules while our proposed involution operator even reduces the complexity of baseline by substituting convolution.

Method	AP^{bbox}	AP^{bbox}_{50}	AP^{bbox}_{75}	AP^{mask}	AP^{mask}_{50}	AP^{mask}_{75}
baseline	38.4	59.2	41.9	35.1	56.3	37.3
+ NL [16]	39.0	61.1	41.9	35.5	58.0	37.4
+ RCCA [7]	39.3	-	-	36.1	-	-
+ GC @C5 [2]	38.7	61.1	41.7	35.2	57.4	37.4
+ DCN @C5 [20]	39.9	-	-	34.9	-	-
+ DGMN @C5 [19]	40.2	62.0	43.4	36.0	58.3	38.2
ours	**40.8**	**62.3**	**44.3**	**36.4**	**59.0**	**38.5**

Table 1: Quantitative comparison on the COCO 2017 validation set. Our model could outstrip the previous methods with attention or dynamic add-on, using reduced parameters and computational cost. C5 indicates inserting the considered components at all the 3×3 convolution layers of the last stage (conv5_x) in ResNet-50.

C. Visualization of Segmentation

Based on the semantic FPN [8] framework, we provide some prediction results on the Cityscapes validation set in Figure 1. Without the help of involution, pixels of large objects are usually mistaken as other objects with high similarity. For instance, the wall in the first image example are mostly confused with building by the convolution-based FPN. Some pixels of the bus in the third image example are misclassified as truck or car, distracted by the occlusion of the cyclist. In contrast, our involution-based FPN dissolves these ambiguities by dynamically reasoning in an enlarged spatial range. Also, better consistency of inner pixels of an object is observed in the segmentation results of our method, reaping the benefits of involution.

D. Discussion

The topological connectivity [5, 6, 17, 18] and hyperparameter configurations [3, 12, 14] of convolutional neural networks have undergone rapid evolution, but developing brand new operators attracts little attention for crafting innovative architectures. In this work, we expect to bridge this regret via disassembling the elements of convolution and reassembling them into a more effective and efficient involution. In the meanwhile, one of the current front edges of neural architecture engineering is automatically searching the network structures [1, 9, 11, 21, 22]. Our invention can also fill the pool of search space for most existing Neural Architecture Search (NAS) strategies. In the near future, we are looking forward to discovering more effective involution-equipped neural networks with the help of NAS.

Figure 1: Qualitative comparison of segmentation results on the Cityscapes validation set. Each column represents an image example of urban scene. The first and second row show the original image and ground truth. The last two rows demonstrate the prediction results of baseline and our method, respectively. Highlighted in the yellow boxes are their apparent differences.

References

[1] Han Cai, Ligeng Zhu, and Song Han. ProxylessNAS: Direct neural architecture search on target task and hardware. In *ICLR*, 2019. 2

[2] Yue Cao, Jiarui Xu, Stephen Lin, Fangyun Wei, and Han Hu. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In *ICCV Workshops*, 2019. 1

[3] Zichao Guo, Xiangyu Zhang, Haoyuan Mu, Wen Heng, Zechun Liu, Yichen Wei, and Jian Sun. Single path one-shot neural architecture search with uniform sampling. In *ECCV*, 2020. 2

[4] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 1

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2

[6] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *CVPR*, 2017. 2

[7] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *ICCV*, 2019. 1

[8] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollar. Panoptic feature pyramid networks. In *CVPR*, 2019. 1

[9] Hanxiao Liu, Karen Simonyan, and Yiming Yang. DARTS: Differentiable architecture search. In *ICLR*, 2019. 2

[10] Chao Peng, Tete Xiao, Zeming Li, Yuning Jiang, Xiangyu Zhang, Kai Jia, Gang Yu, and Jian Sun. Megdet: A large mini-batch object detector. In *CVPR*, 2018. 1

[11] Hieu Pham, Melody Guan, Barret Zoph, Quoc Le, and Jeff Dean. Efficient neural architecture search via parameters sharing. In *ICML*, 2018. 2

[12] Ilija Radosavovic, Raj Prateek Kosaraju, Ross Girshick, Kaiming He, and Piotr Dollar. Designing network design spaces. In *CVPR*, 2020. 2

[13] Prajit Ramachandran, Niki Parmar, Ashish Vaswani, Irwan Bello, Anselm Levskaya, and Jon Shlens. Stand-alone self-attention in vision models. In *NeurIPS*, 2019. 1

[14] Mingxing Tan and Quoc Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *ICML*, 2019. 2

[15] Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. In *ECCV*, 2020. 1

[16] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2018. 1

[17] Saining Xie, Alexander Kirillov, Ross Girshick, and Kaiming He. Exploring randomly wired neural networks for image recognition. In *ICCV*, 2019. 2

[18] Yibo Yang, Zhisheng Zhong, Tiancheng Shen, and Zhouchen Lin. Convolutional neural networks with alternately updated clique. In *CVPR*, 2018. 2

[19] Li Zhang, Dan Xu, Anurag Arnab, and Philip H.S. Torr. Dynamic graph message passing networks. In *CVPR*, 2020. 1

[20] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *CVPR*, 2019. 1

[21] Barret Zoph and Quoc Le. Neural architecture search with reinforcement learning. In *ICLR*, 2017. 2

[22] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V. Le. Learning transferable architectures for scalable image recognition. In *CVPR*, 2018. 2