# Joint COCO and Mapillary Workshop at ICCV 2019: COCO Instance Segmentation Challenge Track Technical Report: Boundary-Aware Localization with Content-Aware Feature Aggregation

Jiaqi Wang<sup>1</sup> Wenwei Zhang<sup>2</sup> Yuhang Cao<sup>1</sup> Kai Chen<sup>1</sup> Jiangmiao Pang<sup>3</sup> Tao Gong<sup>4</sup> Jianping Shi<sup>5</sup> Chen Change Loy<sup>2</sup> Dahua Lin<sup>1</sup> <sup>1</sup>The Chinese University of Hong Kong <sup>2</sup>Nanyang Technological University <sup>3</sup>Zhejiang University <sup>4</sup> University of Science and Technology of China <sup>5</sup> SenseTime Group Limited {wj017,dhlin}@ie.cuhk.edu.hk {yhcao6,chenkaidev,pangjiangmiao,gongtao950513}@gmail.com {wenwei001,ccloy}@ntu.edu.sg shijianping@sensetime.com

# Abstract

In this report, we provide the technical details about our method participating for the COCO Instance Segmentation Challenge Track. First, we revisit the current approach for learning feature pyramid in FPN. We found that the fusion procedure is a crucial factor for fully exploiting multi-scale features extracted in FPN. To improve content-aware fusion, we propose a new Content-Aware *Feature* Aggregation module to enhance the aggregation of pyramidal features in the bottom-up and top-down paths. In addition, a novel module called Decoupled Boundary-Aware Localization is proposed to locate objects more accurately. In particular, we reformulate object localization as a task to localize the four edges of the bounding box for each object, and design a novel pipeline to perform coarse estimation and fine regression. Our overall system achieves 51.3% mask mAP on the COCO test-dev split, without using external instance-level annotated data during training. Code and models will be available at https: //github.com/open-mmlab/mmdetection.

# 1. Methodology

## 1.1. Content-Aware Feature Aggregation (CAFA)

Feature pyramids significantly boost the ability to detect multi-scale objects. Previous methods [12, 12, 14, 11, 17, 5] mainly focus on the information pathway among pyramidal features, while the feature aggregation remains less studied. These works share a similar scheme, *i.e.*, firstly adopt the NN/bilinear upsampling operator to upsample the low-resolution feature map, and then use element-wise sum to fuse it with a high-resolution one. To improve the content-



Figure 1: Modification by CAFA on PAFPN [14]. We use CARAFE [19] and DCNv2 [21] during upsampling and use DCNv2 [21] for downsampling.

awareness duing the fusion process, we explore a new feature fusion manner, named Content-Aware Feature Aggregation (CAFA).

We illustrate the detailed architecture of CAFA based on PAFPN in Figure 1. In the top-down pathway, we replace the upsampling operator with CARAFE [19] to enlarge the receptive field and handle content-aware neighbor regions. To further exploit the semantic information and capture more contextual infomation, we add a modulated deformable convolution (DCNv2) [21] after the upsampling. Adopting CARAFE and DCNv2 makes the feature fusion better adaptive to instance-specific contents. In the bottom-up pathway, we replace the stride-2 convolution with a stride-2 DCNv2 to downsample the feature maps. This design helps to improve the descrimination of downsampled feature, which is more complementary for the lowresolution, semantically strong features.

We conduct extensive experiments with CAFA on Reti-

naNet [13], Mask R-CNN [7] and Hybrid Task Cascade [2] with different backbones. It achieves consistent improvements over the baseline. Notably, CAFA with PAFPN achieves comparable results to NAS-FPN on RetinaNet, which incorporates complicated connections among different feature maps and stacks the pyramid for seven times.

#### **1.2. Decoupled Boundary-Aware Localization**

Accurate object localization is crucial for object detection and instance segmentation. Most recent methods directly regress the normalized deltas between ground-truth boxes and proposals. However, this paradigm may not reach a satisfying localization at one time. Some methods [1, 10] attempt to improve localization performance by cascading pipeline which brings considerable costs. Considering these issues, a Decoupled Boundary-Aware Localization (DBAL) pipeline is proposed to locate the objects more accurately.

As shown in Fig. 2, we localize four boundaries of a bounding box respectively with boundary-specific features. We design a coarse-to-fine localization pipeline where coarse boundaries are first estimated by a boundary classifier, and more precise locations are then obtained by regression. Moreover, the confidence of the estimated boundaries could be also used to represent the reliably of the predicted locations. We therefore further adopts the boundary confidence to assist the object classification. More accurate objects will obtain higher classification confidence, making the NMS procedure keeps the best candidate that has both high classification confidence and accurate localizations.

We first extract the feature of a bounding box with RoI Align. The  $k \times k$  RoI feature is then compressed into 2 1-D feature columns, along the Y-axis and X-axis respectively. These feature columns are boundary-specific representations for the RoI. They split the candidate region into multiple stripes, and each strip is responsible for estimating the object boundary located inside it. Given the extracted RoI feature  $\mathcal{F}_0$ , DBAL first forwards it through two  $3 \times 3$ convolution layers to get the transformed feature  $\mathcal{F}$ . Two attention masks  $\mathcal{M}_x$ ,  $\mathcal{M}_y$  are predicted from  $\mathcal{F}$  by a 1x1 convolution layer. They are normalized by across Y-axis and X-axis respectively. With such attention masks, we can aggregate  $\mathcal{F}$  across Y-axis and X-axis to attain feature for horizontal feature column  $\mathcal{F}_x$  and vertical feature column  $\mathcal{F}_y$ :

$$\mathcal{F}_x = \sum_y \mathcal{F}(y,:) * \mathcal{M}_x(y,:)/k.$$
  
$$\mathcal{F}_y = \sum_x \mathcal{F}(:,x) * \mathcal{M}_y(:,x)/k.$$
 (1)

 $\mathcal{F}_x$  and  $\mathcal{F}_y$  are then refined by a 1-D convolution layer and upsampled into size of  $1 \times 2k$  and  $2k \times 1$ , then decoupled to  $\mathcal{F}_{left}$ ,  $\mathcal{F}_{right}$ ,  $\mathcal{F}_{top}$ ,  $\mathcal{F}_{down}$ . To get the coarse regression results, each feature element on  $\mathcal{F}_{left}$ ,  $\mathcal{F}_{right}$ ,  $\mathcal{F}_{top}$ ,  $\mathcal{F}_{down}$  predicts the confidence of whether it is nearest to the object boundary independently. A refined regression is performed based on the estimation results leveraging the feature on the coarse location.

Due to the non-precise localization of the proposal box, the proposal is rescaled by a factor of  $\sigma > 1$ to cover the whole object during calculating the regress targets. It means that the RoI feature is pooled with original proposal box ( $B_{left}, B_{right}, B_{top}, B_{down}$ ), and the regress target is calculated with rescale box ( $\sigma B_{left}, \sigma B_{right}, \sigma B_{top}, \sigma B_{down}$ ). The RoI Feature from a deep network providing a large receptive field to cover the rescaled box region, a similar problem is also discussed in [15]. The average boundary confidence is adjusted by the average boundary confidence score in the final step. Experimental results show substantial gains by this simple classification rescoring step.

#### 2. Experiments

# 2.1. Ablation Study

**Content-Aware Feature Aggregation (CAFA).** We study the effectiveness of each component in the CAFA combined with FPN [12] and PAFPN [14] baseline in Mask R-CNN [7] as shown in Table 1. CAFA brings 1.1% and 1.9% mask AP gains on FPN and PAFPN, respectively. The results of fourth and fifth rows also suggest that combining CARAFE with DCNv2 could further improves the results. And appling DCNv2 after CARAFE is 0.7% mask AP better than applying DCNv2 before CARAFE in PAFPN [14]. As shown in the second row in both FPN and PAFPN subtable, replacing the output convolution by DCNv2 in FPN and PAFPN brings less improvements. It indicates adopting the content-aware operation (*e.g.*, DCNv2) before fusion could more effectively enhance the features than adopting these operations in a straight forward path.

We further compare CAFA with NAS-FPN on RetinaNet and achieve compatible results (box AP 39.2% v.s. 39.5% on COCO2017 val dataset at  $640 \times 640$  scale). While NAS-FPN uses 7 pyramid networks, our CAFA with PAFPN only uses 2 pyramid networks with much simpler pathways (one top-down and one bottom-up) among pyramidal features. This comparison proves the effectiveness of CAFA.

**Decoupled Edge-Aware Localization (DBAL)**. We evaluate Decoupled Edge-Aware Localization (DBAL) on Faster R-CNN and Cascade R-CNN with both ResNet-50 backbone and 1x training scheduler. DBAL shows significant performance gains on both methods. To be specific, DBAL improves Faster R-CNN by 3.4% box AP and Cascade R-CNN by 1.1% box AP. We further evaluate the effectiveness of rescoring the classification results by the location estimation confidence, which is described in Sec. 2. The rescoring mechanism shows performance gains (0.8% and 0.1%) based on strong performances (39.0% and 41.4%).



Figure 2: Structure of Decoupled Boundary-Aware Localization (DBAL).

Table 1: Effectiveness of CARAFE [19] and DCNv2 [21] in FPN [12] and PAFPN [14] in Mask R-CNN [7].

Method	Modification	box AP	mask AP
FPN	Baseline	37.3	34.2
	+ DCNv2@out	37.8	34.4
	+ CARAFE	38.1	34.8
	+ DCNv2 + CARAFE	38.8	35.2
	+ CARAFE + DCNv2 (CAFA)	38.9	35.3
	Baseline	37.7	34.3
PAFPN	+ DCNv2@out	38.5	34.9
	+ CARAFE	38.8	35.4
	+ DCNv2 + CARAFE	39.4	35.5
	+ CARAFE + DCNv2 (CAFA)	40.0	36.2

Table 2: Effectiveness of DBAL in Faster R-CNN and Cascade R-CNN on COCO2017 *val* dataset.

Method	Modification	box AP	
Faster P CNN	Baseline	36.4	
Faster K-CININ	+ DBAL w/o rescoring	39.0	
	+ DBAL	39.8	
Cascada P. CNN	Baseline	40.4	
Cascade K-CININ	+ DBAL w/o rescoring	41.4	
	+ DBAL	41.5	

#### 2.2. Extensions

After applying DBAL to Mask R-CNN [7], we achieve 35.0% mask AP. Then we apply it to the Hybrid Task Cascade (HTC) [2] with the proposed CAFA in PAFPN [14] and CARAFE [19] in Mask Head. With ResNet-50 [8] backbone and 1x training scheduler [6], our method achieves 38.4% mask AP and 44.3% box AP compared with 37.3% mask AP and 42.1% box AP HTC baseline. Our overall system is trained without involving external instance-level annotated data during training. To be specific, it is trained on COCO2017 training split (instance segmentation and stuff annotations) as in [2]. Here we also list

Table 3: Step by Step results (bbox AP& mask AP) of our method on COCO2017 *val* dataset.

Methods	scheduler	AP <sub>box</sub>	$AP_{mask}$
Mask R-CNN	1x	37.3	34.2
+ DBAL	1x	40.0 (+2.7)	35.0 (+0.8)
+ HTC	1x	42.9 (+2.9)	37.4 (+2.4)
+ CAFA&CARAFE	1x	44.3 (+1.4)	38.4 (+1.0)
+ SyncBN	1x	45.8 (+1.5)	39.9 (+1.5)
+ SW	1x	46.1 (+0.3)	40.0 (+0.1)
+ Backbone DCNv2	1x	48.2 (+2.1)	41.7 (+1.7)
+ Mask Scoring	1x	48.3 (+0.1)	42.4 (+0.7)
+ MS-Training	20e	50.2 (+1.9)	44.5 (+2.1)
+ SE154-SW	20e	52.7 (+2.5)	46.1 (+1.6)
+ AutoAug&InstaBoost	4x	54.0 (+1.3)	47.1 (+1.0)
+ Multi-Scale Testing	-	55.3 (+1.3)	48.4 (+1.3)
+ Ensemble	-	57.2 (+1.9)	50.5 (+2.1)

other steps and additional modules we used to obtain the final performance. The step-by-step gains brought by different components are illustrated in Table 3.

**SyncBN**. We use Synchronized Batch Normalization [14, 18] in the backbone and heads.

**SW**. We adopt Switchable Whitening (SW) [16] in the backbone and FPN following the original paper.

**DCNv2**. We appply Deformable Convolution v2 [21] in the last three stage (from res3 to res5) of the backbone.

**Multi-scale Training.** We adopt multi-scale training. The scale of short edge is randomly sampled from [400, 1400] per iteration and the scale of long edge is fixed as 1600. The detectors are trained with 20 epoches and the learning rate is decreased by 0.1 after 16 and 19 epoches, respectively.

**SENet-154 with SW**. We tried different bigger backbones. SENet-154 [9] with Switchable Whitening (SW) [16] achieves the best single model performance.

**Stronger Augmentation**. We adopt Instaboost [4] as the sixth policy of AutoAugment [3]. Each policy has the same

Table 4: Results (mask AP) with better backbones and bells and whistles on COCO2017 *test-dev* dataset.

Methods		$AP_{50} \\$	$AP_{75} \\$	$AP_S$	$AP_M$	$AP_L$
2018 Winners Single Model		70.6	52.1	30.2	50.1	61.8
Ours Single Model		72.0	54.3	29.0	51.5	66.6
2018 Winners Ensemble [2]	49.0	73.0	53.9	33.9	52.3	61.2
Ours	51.3	74.7	56.5	30.6	53.5	68.9

probability to be used for data augmentation during training procedure. The detectors are trained with 48 epoches with such stronger augmentation, and the learning rate is decreased by 0.1 after 40 and 45 epoches, respectively.

**Multi-scale Testing**. We use 5 scales as well as horizontal flip at test time before ensemble. The testing scales are (600, 900), (800, 1200), (1000, 1500), (1200, 1800), (1400, 2100).

**Ensemble**. We use ensemble of models based on five backbone networks. We pretrain SENet-154 w/ SW and SE-ResNext-101 w/ SW on ImageNet-1K image classification dataset and use pretrained weights of ResNeXt-101  $32 \times 32d$ , ResNeXt-101  $32 \times 16d$  [20] and ResNeXt-101  $32 \times 8d$  [20] provided by PyTorch<sup>1</sup>.

On COCO 2017 test-dev dataset, our method finally achieves 51.3% Mask AP with multiple model ensemble and 49.4% Mask AP with single model. Our result outperforms the 2018 COCO Winner Entry by 2.3% Mask AP.

# **3.** Conclusion

In conclusion, we propose **Content-Aware Feature Aggregation** (CAFA) to further enhance the pyramidal feature representation learning and **Decoupled Edge-Aware Localization** (DEAL) to replace the conventional bounding box regression. With these two contributions, we achieve new state-of-the-art on the COCO Instance Segmentation Challenge, and remarkably (**2.3**% mask AP) surpass the 2018 winner results.

## References

- Zhaowei Cai and Nuno Vasconcelos. Cascade R-CNN: delving into high quality object detection. In *IEEE Conference* on Computer Vision and Pattern Recognition, 2018. 2
- [2] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. Hybrid task cascade for instance segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, June 2019. 2, 3, 4
- [3] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation strategies from data. In *CVPR*, June 2019. 4
- [4] Haoshu Fang, Jianhua Sun, Runzhong Wang, Minghao Gou, Yonglu Li, and Cewu Lu. Instaboost: Boosting instance segmentation via probability map guided copy-pasting. *CoRR*, abs/1908.07801, 2019. 3

- [5] Golnaz Ghiasi, Tsung-Yi Lin, Ruoming Pang, and Quoc V. Le. NAS-FPN: learning scalable feature pyramid architecture for object detection. *CoRR*, abs/1904.07392, 2019. 1
- [6] Ross Girshick, Ilija Radosavovic, Georgia Gkioxari, Piotr Dollár, and Kaiming He. Detectron. https://github. com/facebookresearch/detectron, 2018. 3
- [7] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *IEEE International Conference* on Computer Vision, 2017. 2, 3
- [8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, Jun 2016. 3
- [9] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, Jun 2018. 3
- [10] Borui Jiang, Ruixuan Luo, Jiayuan Mao, Tete Xiao, and Yuning Jiang. Acquisition of localization confidence for accurate object detection. In *European Conference on Computer Vi*sion, 2018. 2
- [11] Tao Kong, Fuchun Sun, Chuanqi Tan, Huaping Liu, and Wenbing Huang. Deep feature pyramid reconfiguration for object detection. In *European Conference on Computer Vi*sion, 2018. 1
- [12] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 1, 2, 3
- [13] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision*, 2017. 2
- [14] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 1, 2, 3
- [15] Xin Lu, Buyu Li, Yuxin Yue, Quanquan Li, and Junjie Yan. Grid r-cnn. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019. 2
- [16] Xingang Pan, Xiaohang Zhan, Jianping Shi, Xiaoou Tang, and Ping Luo. Switchable whitening for deep representation learning. *CoRR*, abs/1904.09739, 2019. 3
- [17] Jiangmiao Pang, Kai Chen, Jianping Shi, Huajun Feng, Wanli Ouyang, and Dahua Lin. Libra r-cnn: Towards balanced learning for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 1
- [18] Chao Peng, Tete Xiao, Zeming Li, Yuning Jiang, Xiangyu Zhang, Kai Jia, Gang Yu, and Jian Sun. Megdet: A large mini-batch object detector. *IEEE Conference on Computer Vision and Pattern Recognition*, Jun 2018. 3
- [19] Jiaqi Wang, Kai Chen, Rui Xu, Ziwei Liu, Chen Change Loy, and Dahua Lin. CARAFE: content-aware reassembly of features. *CoRR*, abs/1905.02188, 2019. 1, 3
- [20] Saining Xie, Ross Girshick, Piotr Dollar, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *IEEE Conference on Computer Vision* and Pattern Recognition, Jul 2017. 4
- [21] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *IEEE Conference on Computer Vision and Pattern Recognition*, June 2019. 1, 3

<sup>&</sup>lt;sup>1</sup>https://pytorch.org/hub/facebookresearch\_WSL-Images\_resnext/