ACCURATE INSTANCE SEGMENTATION VIA COLLABORATIVE LEARNING

*Tianyou Chen[1], Xiaoguang Hu[1], Jin Xiao[*1], Guofeng Zhang[1], Shaojie Wang[1]*

[1]State Key Laboratory of Virtual Reality Technology and Systems, School of Automation Science and Electrical Engineering, Beihang University Beijing 100191, China

ABSTRACT

We propose an instance segmentation model, named CoMask, that effectively alleviates the scale variation issue and addresses the precise localization. Specifically, we develop a multi-scale feature extraction module (MSFEM) to exploit multi-scale spatial cues. Besides, the channel attention mechanism is also adopted to further enhance the discriminating ability. Equipped with MSFEMs, multi-scale and multi-level features can be extracted to better characterize objects of various sizes and provide affluent high-level semantic information. For precise localization, we propose a collaborative learning framework to compute coarse masks and regresses position-sensitive dense offsets. The foreground confidence of each position is then assigned as the weight of the corresponding bounding box to calculate a weighted average. Thus, we can mitigate interference of background regions. After obtaining the final regressed bounding boxes, finer foreground masks can be calculated. We conduct experiments on MS COCO dataset. Experimental results validate that CoMask is competitive compared with state-of-the-art models.

Index Terms— Collaborative learning, deep learning, instance segmentation, multi-scale feature extraction

1. INTRODUCTION

Instance segmentation aims at simultaneously detecting all objects in an image and accurately segmenting each instance. As a fundamental computer vision task, instance segmentation has aroused tremendously increasing research interest [1]-[5]. Despite the extraordinary performance of these models, there are still several problems remaining unsolved.

First, as pointed out in [6], a single-level feature extracted from a convolutional neural network (CNN) can only capture the scale-specific information. However, object instances are of various scales, which presents an extreme challenge to CNN-based methods [7]. Previous methods [1], [8], [9] solve this problem by detecting different sizes of instances on different levels of side-outputs. Thus, small proposals are assigned to low-level features and large proposals are assigned to high-level ones. Albeit simple and effective, as demonstrated in [10], this strategy may generate suboptimal results. Besides, recent studies [11], [12] have proven that the effective receptive field size of a CNN is much small than its theoretical value. Thus, existing methods have difficulty in modeling larger context.

Second, previous instance segmentation models mainly rely on predefined anchors and employ several fully connected layers to compute a single bounding box of the candidate proposal, which cannot fully leverage the foreground samples to train the regressor [13]. Recently, many methods [4], [14] are proposed to solve this problem and have shown better performance compared with anchor-based counterparts. However, these methods still follow the detect-then-segment scheme. Although this framework is beneficial to the mask computation, it provides little help to the object detection, which may lead to suboptimal results.

In this paper, we propose a novel network named CoMask to remedy the aforementioned challenges. First, to exploit multi-scale spatial cues and enhance the single-level representation, we propose the multi-scale feature extraction module (MSFEM). The MSFEM comprises multiple sub-branches and different sub-branches contain different number of convolutional layers, which is not only effective in capturing multi-scale information but also helpful to enlarge the effective receptive field of the network. To further enhance the discriminating ability, we embed a channel attention module into the MSFEM to help the proposed CoMask focus more on foreground regions. The MSFEMs are then plugged into the feature pyramid network (FPN) to build a multi-scale feature pyramid network (MFPN). Thus, the segmentation accuracy can be improved with limited speed overhead.

Besides, a novel collaborative learning framework is proposed to achieve better performance. In collaborative learning, several group members work together to realize the learning goals via exploratory learning and timely interaction [15]. In our framework, two mutually beneficial collaborators (i.e., object detection and mask segmentation) are designed. On the one hand, we follow many state-of-the-art methods and adopt the two-stage paradigm. After obtaining proposals derived from the region proposal network, we compute coarse segmentation masks by using a light-weight network. On the other hand, inspired by the FCOS [13], we employ the RoI features to compute the position-sensitive dense offsets. Thus, we can take better advantage of the foreground samples to train the regressor. Considering that bounding boxes produced by locations belonging to background regions may be of low quality, we innovatively integrate the object detection and mask segmentation in a mutually beneficial manner. Specifically, after obtaining the regressed bounding boxes, we assign different weights to different boxes to calculate a weighted average. Thus, we can avoid the interference of background regions on the final box regression. After obtaining the final bounding box, we can compute the final segmentation masks.

To demonstrate the excellent performance of the CoMask and evaluate the effectiveness of the proposed MSFEM and collaborative learning framework, we conduct extensive

[*] Corresponding author

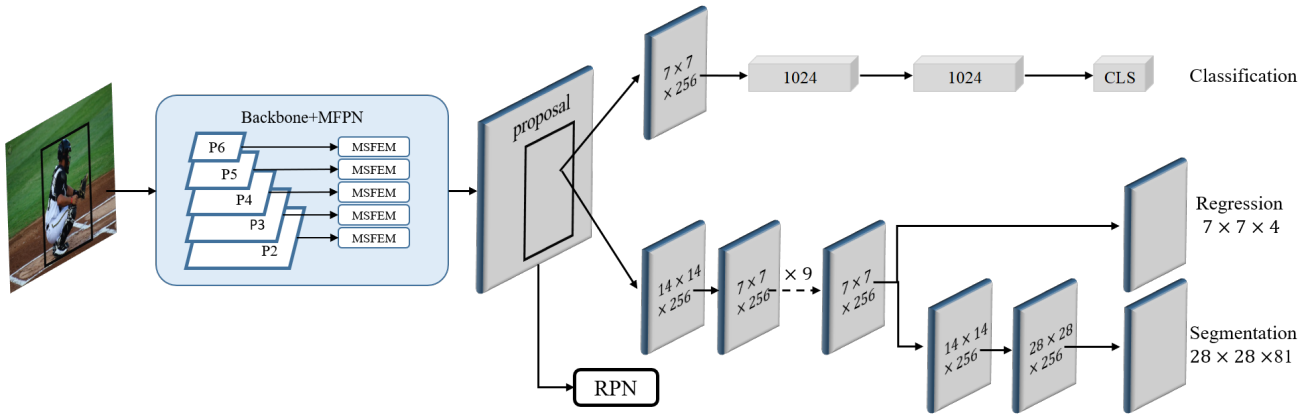

Fig. 1. Overall architecture of the proposed CoMask. $\{\boldsymbol{P_2}, \ldots, \boldsymbol{P_6}\}$ are the output feature maps of the FPN.

experiments on the MS COCO dataset. Experimental results validate that CoMask is able to generate high-quality segmentation masks and precisely locate the object instances. In summary, this paper makes the following major contributions:

- We propose the MSFEM to exploit multi-scale spatial cues and enlarge the effective receptive field. A channel attention block is plugged into the MSFEM to further enhance the discriminating ability. Equipped with MSFEMs, the proposed CoMask can better solve the scale variation issue.
- We propose a collaborative learning framework where object detection and mask segmentation are integrated in a mutually beneficial manner. Thus, we can fully leverage the foreground samples to facilitate the training of the regressor and mitigate the interference of background regions.
- Extensive experimental results on the MS COCO dataset prove that the CoMask is competitive compared with state-of-the-art methods.

2. RELATED WORKS

Object detection: Recently, it has been witnessed prospering researches on object detection, which can be attributed to the dramatic advances in deep learning. Ren et al. propose a two-stage model named Faster R-CNN [8] for object detection. In the first stage, Faster R-CNN employs the region proposal network (RPN) to compute class-agnostic region proposals. In the second stage, a Fast R-CNN detector [16] is utilized to calculate the classification scores of each proposal and regress the corresponding bounding box. Lin et al. [9] design a feature pyramid network (FPN) to generate features having affluent semantic information at all scales. Redmon et al. present a novel one-stage model named YOLO [17], [18], where the bounding box coordinates and the class probabilities can be straightly derived from image pixels. Liu et al. [19] propose the single shot multibox detector (SSD), where default boxes of different scales are distributed to different levels to capture objects of various sizes. Lin et al. [20] propose the focal loss to solve the foreground-background class imbalance issue. Tian et al. [13] propose an anchor box free detection framework, which largely simplifies the detection pipeline and reduces the computation overhead.

Instance segmentation: Instance segmentation aims at locating objects of interest at the pixel-level and is closely related to object detection. Currently, state-of-the-art instance segmentation methods mainly solve this problem by extending object detection models. Thus, the foreground masks can be obtained by predicting pixels on each bounding box. He et al. [1] extends the Faster R-CNN by adding a mask prediction branch. Lee et al. [4] plug a spatial attention-guided mask branch into the FCOS detector in the same way with Mask R-CNN. Chen et al. [21] introduce cascade to instance segmentation to learn discriminative features progressively. Liu et al. [10] propose a bottom-up path augmentation strategy to boost the segmentation performance. Wang et al. [22] convert instance segmentation to a simple classification-solvable problem and propose a much flexible and simpler instance segmentation model.

3. PROPOSED METHOD

The overall architecture of the proposed CoMask is illustrated in Fig. 1. Given an input image, we first use the MFPN to extract multi-level and multi-scale features. Similar to other two-stage models, a standard RPN is leveraged to obtain proposals, which are then utilized to calculate the bounding boxes and segmentation masks.

3.1 MSFEM

The structure of the proposed MSFEM is shown in Fig. 2. As demonstrated in the figure, each module contains four sub-branches and different sub-branches have different number of convolutional layers. Concretely, we use 3×3 convolutional layers with a stride of 2 to capture multi-scale information. The process can be formulated as:

$$F_i = C_2^i(F), i = 0,1,2,3, \tag{1}$$

where F is the input feature map, C_2 denotes a 3×3 convolutional layer with a stride of 2, C_2^i is an iterated function and $C_2^i(F) = C_2\left(C_2^{i-1}(F)\right)$. It worth noting that $F_0 = C_2^0(F) = F$. These multi-scale feature maps are then integrated via channel-wise concatenation. The fused feature is defined as:

$$F_F = cat(F_0, F_1, F_2, F_3), \tag{2}$$

where cat is the concatenation operation. To further enhance the discriminating ability, we plug the channel attention into the MSFEM. Specifically, we first use a 1×1 convolutional layer to reduce the channel number of F_F. Then, a global average pooling

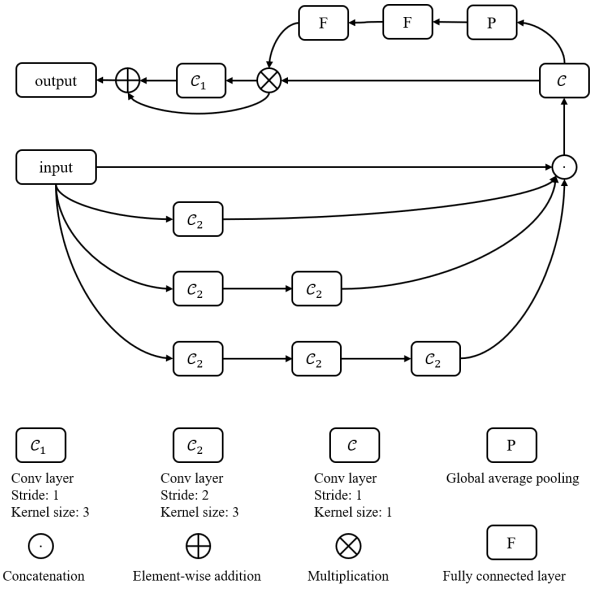

Fig. 2. Structure of the MSFEM.

layer is utilized to obtain a one-dimension vector, which is fed to two consecutive fully connected layers to exploit channel-wise dependencies. The process can be formulated as:

$$F_f = Conv(F_F), \tag{3}$$

$$V_1 = gp(F_f), \tag{4}$$

$$V_2 = \sigma\left(fc_1\left(fc_2(V_1)\right)\right), \tag{5}$$

where *Conv* is a 1×1 convolutional layer, *gp* denotes the global average pooling, fc_1 and fc_2 are fully connected layers, σ represents the sigmoid function. We obtain the output of MSFEM by computing:

$$\widehat{F_f} = F_f \times EX(V_2; F_f), \tag{6}$$

$$F_O = C_1(\widehat{F_f}), \tag{7}$$

where $EX(V_2; F_f)$ is a function utilized to expand the vector V_2 to the same size as F_f, C_1 denotes a 3×3 convolutional layer with a stride of 1, F_O is the output.

We build the MFPN by adding five MSFEMs after the FPN at five feature levels. Generally speaking, the addition of MSFEMs brings us two benefits: 1) multi-scale spatial cues can be exploited; 2) effective receptive field can be enlarged. It worth noting that in each MSFEM, a half of the convolution operations are conducted on subsampled feature maps. Thus, the addition of MSFEMs will not introduce significant computation overhead.

3.2 Collaborative learning

Similar to other two-stage methods, we first use an RPN to obtain proposals. Inspired by the FCOS [13] and D2Det [5], we perform object detection in pixel-wise prediction fashion. Concretely, given a candidate object proposal, we first employ the RoIAlign [1] to obtain the 14×14 RoI feature. Then, to reduce the memory and computation overhead, we use a 3×3 convolutional layer with a stride of 2 to reduce the spatial resolution. Nine stacked convolutional layers are then leveraged for feature refinement. We use two stacked deconvolutional layers and a single convolutional layer with an output channel number of 81 to generate coarse segmentation masks. Meanwhile, a convolutional layer is utilized to generate the dense bounding box offsets. Let (l_i, t_i, r_i, b_i) be the bounding box predicted by the pixel with coordinates (x_i, y_i), and M be the corresponding foreground mask. We apply an average pooling layer to the coarse segmentation mask to obtain mask M_s with a spatial resolution of 7×7. Then, the foreground confidence of pixel (x_i, y_i) can be denoted as $M_s(x_i, y_i)$. Similar to other methods, we only consider pixels with $M_s(x_i, y_i) > 0.5$. The final bounding box (l, t, r, b) can be obtained by computing:

$$l = \frac{\sum l_i \times M_s(x_i, y_i)}{\sum M_s(x_i, y_i)}, t = \frac{\sum t_i \times M_s(x_i, y_i)}{\sum M_s(x_i, y_i)}, \tag{8}$$

$$r = \frac{\sum r_i \times M_s(x_i, y_i)}{\sum M_s(x_i, y_i)}, b = \frac{\sum b_i \times M_s(x_i, y_i)}{\sum M_s(x_i, y_i)}, \tag{9}$$

Afterwards, we can use the final bounding box to extract more accurate RoI feature. The extracted RoI feature is then fed into the segmentation head to obtain finer segmentation mask.

4. EXPERIMENTS

4.1. Dataset and implementation details

Dataset and evaluation metrics: We conduct extensive experiments on MS COCO [27] benchmark dataset. As the most widely used dataset in instance segmentation and object detection, MS COCO dataset involves 80 object categories and is divided into three subsets: trainval, minival, and test-dev. For fair comparison, we perform training on the trainval subset and report the final results on the test-dev subset as done in many state-of-the-art [1]-[5] methods. We adopt the widely used average precision (AP) to evaluate the overall performance, which is measured using mask Intersection-over-Union (IoU) between the predicted mask and the corresponding groundtruth mask. Additionally, we report AP, AP_{50}, AP_{75}, and AP_S, AP_M, AP_L to provide more comprehensive evaluation. The AP is calculated by utilizing 10 IoU thresholds ranging from 0.5 to 0.95 with an interval of 0.05. AP_{50} and AP_{75} are computed by using a single IoU threshold of 0.5 and 0.75, respectively. AP_S, AP_M, AP_L indicate the AP of small (area < 32^2), medium (32^2 < area < 96^6) and large (area > 96^6) objects.

Implementation Details: Unless specified, we employ ResNet-50 [28] as the backbone network. Following previous methods [4], [5], [13], each input image is resized to have a shorter side = 800 pixels and a longer side ≤ 1333 pixels. The CoMask is trained on 4 NVIDIA Titan Xp GPUs. An SGD optimizer with a weight decay of 1e-4 and a momentum of 0.9 is utilized to train the CoMask. We only use the traditional horizontal flipping for data augmentation and the whole network is trained with a 2× scheme.

4.2. Comparison with state-of-the-art methods

We compare the CoMask with 9 state-of-the-art methods including PANet [10], CondInst [23], BlendMask [3], MS R-CNN [2], Mask R-CNN [1], RetinaMask [25], ShapeMask [26], Cascaded Mask R-CNN [21], and Mask SSD [24]. For fair comparison, we report single-model and single-scale results of all methods. The quantitative experimental results on MS COCO dataset are shown in Table 1. The Mask SSD1024 is a variant of the Mask SSD, which is trained on 1024×1024 images. As is demonstrated in the table,

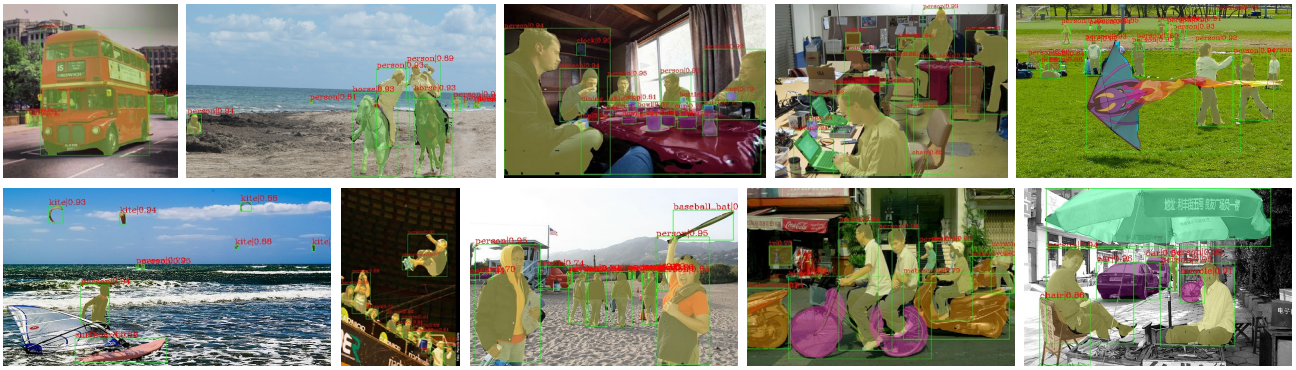

Fig. 3. Qualitative results of CoMask on COCO.

Table 1. Comparison of CoMask and 9 state-of-the-art methods on MS COCO test-dev. For fair comparison, we report the single-model and single-scale results of all methods. The best results are highlighted in **bold**.

Method	Backbone	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
PANet [10]	ResNet-50	36.6	58.0	39.3	16.3	38.1	**53.1**
CondInst [23]	ResNet-50	35.4	56.4	37.6	18.4	37.9	46.9
BlendMask [3]	ResNet-50	34.3	55.4	36.6	14.9	36.4	48.9
CoMask	ResNet-50	**37.7**	**59.0**	**40.9**	**21.0**	**40.9**	48.5
MS RCNN [2]	ResNet-101	38.3	58.8	41.5	17.8	40.4	**54.4**
Mask R-CNN [1]	ResNet-101	35.7	58.0	37.8	15.5	38.1	52.4
RetinaMask [25]	ResNet-101	34.7	55.4	36.9	14.3	36.7	50.5
ShapeMask [26]	ResNet-101	37.4	58.1	40.0	16.1	40.1	53.8
Cascaded Mask R-CNN [21]	ResNet-101	38.4	**60.2**	41.4	20.2	41.0	50.6
Mask SSD1024 [24]	ResNet-101	33.1	53.1	35.0	12.8	34.9	59.0
CoMask	ResNet-101	**38.6**	60.1	**41.9**	**21.2**	**41.9**	50.3

our CoMask outperforms other approaches and achieves an AP score of 38.6 when using ResNet-101 as the backbone network. In particular, a larger performance gain of 0.5% is obtained at strict evaluation metric (i.e., AP@75), compared with the best competing approach (i.e., Cascaded Mask R-CNN), which validates the effectiveness of the propose CoMask. Additionally, we provide several representative results in Fig. 3 to further demonstrate the prominent performance of the proposed CoMask. As shown in the figure, the CoMask is competent to generate high-quality segmentation masks in various challenging scenarios.

4.3. Ablation studies

We conduct ablation studies to evaluate the effectiveness of the proposed MSFEM and collaborative learning framework. For fair comparison, all variants are built on ResNet-50 backbone network and evaluated on the MS COCO minival subset.

The effectiveness of the MSFEM. We implement three variants of the CoMask to prove the effectiveness of the MSFEM and evaluate the impacts of the number of sub-branches. We use CoMask_i to denote the variant with i-branch MSFEM. As shown in Table 2, CoMask_4 shows the best overall performance, which verifies the effectiveness of the proposed MSFEM. In particular, the improvement is more obvious when detecting large instances, which proves that MSFEM is effective in modeling larger context.

The Effectiveness of the collaborative learning framework. We implement a model (denoted as *w/o* CL) without using the collaborative learning strategy to evaluate the effectiveness. The experimental results are shown in Table 3. As demonstrated in the table, CoMask outperforms *w/o* CL, which validates the effectiveness of the collaborative learning framework.

Table 2. Ablation analysis for the proposed MSFEM. The inference speed of each variant is tested on a single NVIDIA Titan Xp Gpu. The best results are highlighted in **bold**.

Methods	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	FPS
CoMask_1	36.7	57.4	39.6	19.8	40.3	49.1	4.4
CoMask_2	37.2	**58.2**	**40.2**	20.1	**40.9**	49.9	4.2
CoMask_4	**37.3**	**58.2**	**40.2**	**20.2**	40.8	**50.3**	3.9

Table 3. Ablation analysis for the proposed collaborative learning framework. *w/o* CL indicates the variant without using collaborative learning strategy. The best results are highlighted in **bold**.

Method	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
CoMask	**37.3**	**58.2**	**40.2**	20.2	**40.8**	**50.3**
w/o CL	37.0	**58.2**	39.8	**20.3**	40.5	49.9

5. CONCLUSION

In this paper, we propose a novel CNN-based model named CoMask for instance segmentation. First, we design the MSFEM to extract multi-scale spatial information and enlarge the effective receptive field. Afterwards, we propose a novel collaborative learning framework to integrate object detection and instance segmentation in a mutually beneficial manner. The collaborative learning strategy not only fully leverage foreground samples to facilitate the training of the regressor but also effectively mitigate the interference of background regions. Experimental results on the MS COCO benchmark dataset demonstrate that CoMask is competitive compared with 9 state-of-the-art methods.

6. REFERENCES

[1] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," In ICCV, 2017, pp. 2980–2988, doi: 10.1109/ICCV.2017.322.

[2] Z. Huang, L. Huang, Y. Gong, C. Huang, and X. Wang, "Mask Scoring R-CNN," In CVPR, 2019, pp. 6402–6411, doi: 10.1109/CVPR.2019.00657.

[3] H. Chen, K. Sun, Z. Tian, C. Shen, Y. Huang, and Y. Yan, "BlendMask: Top-Down Meets Bottom-Up for Instance Segmentation," In CVPR, 2020, pp. 8570–8578, doi: 10.1109/CVPR42600.2020.00860.

[4] Y. Lee and J. Park, "CenterMask: Real-Time Anchor-Free Instance Segmentation," In CVPR, 2020, pp. 13903–13912, doi: 10.1109/CVPR42600.2020.01392.

[5] J. Cao, H. Cholakkal, R. M. Anwer, F. S. Khan, Y. Pang, and L. Shao, "D2Det: Towards High Quality Object Detection and Instance Segmentation," In CVPR, 2020, pp. 11482–11491, doi: 10.1109/CVPR42600.2020.01150.

[6] T. Chen, X. Hu, J. Xiao, and G. Zhang, "BPFINet: Boundary-aware Progressive Feature Integration Network for Salient Object Detection," *Neurocomputing*, 2021, doi: https://doi.org/10.1016/j.neucom.2021.04.078.

[7] B. Singh and L. S. Davis, "An Analysis of Scale Invariance in Object Detection - SNIP," In CVPR, 2018, pp. 3578–3587, doi: 10.1109/CVPR.2018.00377.

[8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, 2017, doi: 10.1109/TPAMI.2016.2577031.

[9] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," In CVPR, 2017, doi: 10.1109/CVPR.2017.106.

[10] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path Aggregation Network for Instance Segmentation," In CVPR, 2018, pp. 8759–8768, doi: 10.1109/CVPR.2018.00913.

[11] S. Li, X. Sui, X. Luo, X. Xu, Y. Liu, and R. Goh, "Medical Image Segmentation Using Squeeze-and-Expansion Transformers," In IJCAI, pages 807-815, 2021.

[12] Y. Mao *et al.*, "Transformer Transforms Salient Object Detection and Camouflaged Object Detection," vol. 14, no. 8, pp. 1–15, 2021, [Online]. Available: http://arxiv.org/abs/2104.10127.

[13] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully Convolutional One-Stage Object Detection," In ICCV, 2019, pp. 9626–9635, doi: 10.1109/ICCV.2019.00972.

[14] X. Zhou, J. Zhuo, and P. Krähenbühl, "Bottom-Up Object Detection by Grouping Extreme and Center Points," In CVPR, 2019, pp. 850–859, doi: 10.1109/CVPR.2019.00094.

[15] W. Ji, J. Li, M. Zhang, Y. Piao, and H. Lu, "Accurate RGB-D Salient Object Detection via Collaborative Learning," In ECCV, 2020, doi: 10.1007/978-3-030-58523-5_4.

[16] R. Girshick, "Fast R-CNN," In ICCV, 2015, pp. 1440–1448, doi: 10.1109/ICCV.2015.169.

[17] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," In CVPR, 2016, pp. 779–788, doi: 10.1109/CVPR.2016.91.

[18] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," In CVPR, 2017, doi: 10.1109/CVPR.2017.690.

[19] W. Liu et al., "SSD: Single shot multibox detector," In ECCV, 2016, pp. 21–37, doi: 10.1007/978-3-319-46448-0_2.

[20] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal Loss for Dense Object Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, 2020, doi: 10.1109/TPAMI.2018.2858826.

[21] K. Chen *et al.*, "Hybrid Task Cascade for Instance Segmentation," In CVPR, 2019, pp. 4969–4978, doi: 10.1109/CVPR.2019.00511.

[22] X. Wang, T. Kong, C. Shen, Y. Jiang, and L. Li, "SOLO: Segmenting Objects by Locations," In ECCV, 2020, pp. 649–665.

[23] Z. Tian, C. Shen, and H. Chen, "Conditional Convolutions for Instance Segmentation," In ECCV, 2020, pp. 282–298.

[24] H. Zhang, Y. Tian, K. Wang, W. Zhang, and F.-Y. Wang, "Mask SSD: An Effective Single-Stage Approach to Object Instance Segmentation," *IEEE Trans. Image Process.*, vol. 29, pp. 2078–2093, 2020, doi: 10.1109/TIP.2019.2947806.

[25] C.-Y. Fu, M. Shvets, and A. C. Berg, "RetinaMask: Learning to predict masks improves state-of-the-art single-shot detection for free," 2019, [Online]. Available: http://arxiv.org/abs/1901.03353.

[26] W. Kuo, A. Angelova, J. Malik, and T. Y. Lin, "ShapeMask: Learning to segment novel objects by refining shape priors," In ICCV, 2019, doi: 10.1109/ICCV.2019.00930.

[27] T.-Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context," In ECCV, 2014, pp. 740–755.

[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," In CVPR, 2016, doi: 10.1109/CVPR.2016.90.