FCOS:Fully Convolutional One-Stage Object Detection

Zhi Tian， Chunhua Shen *， Hao Chen， Tong He
The University of Adelaide， Australia

ICCV2019

ObjectBox: From Centers to Boxes for Anchor-Free Object Detection

Mohsen Zand , Ali Etemad , and Michael Greenspan
Dept. of Electrical and Computer Engineering, Ingenuity Labs Research InstituteQueen’s University, Kingston, Ontario, Canada

ECCV 2022 Oral

Wang Xinyu
2022.11.23

FCOS:Fully Convolutional One-Stage Object Detection

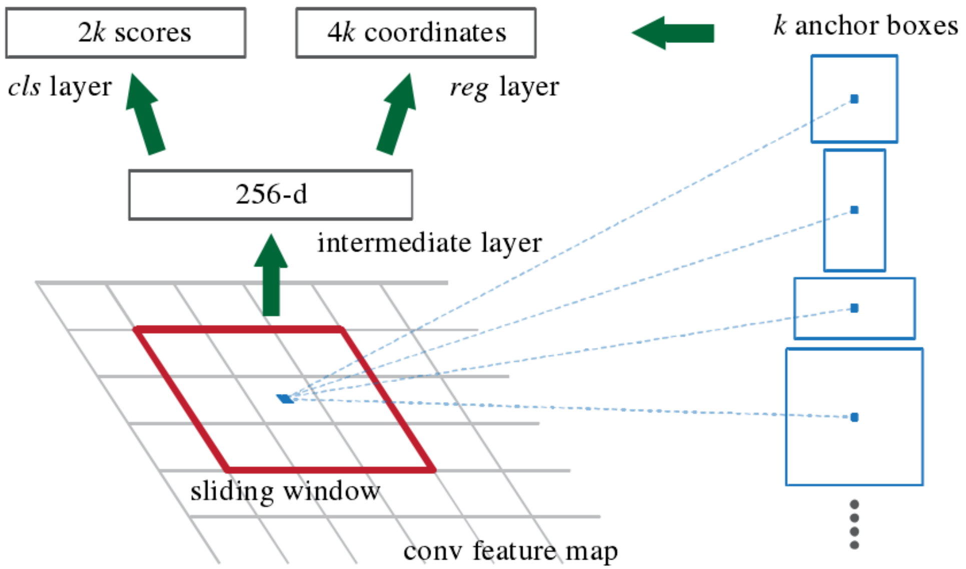
Zhi Tian， Chunhua Shen * ， Hao Chen ， Tong He
The University of Adelaide， Australia

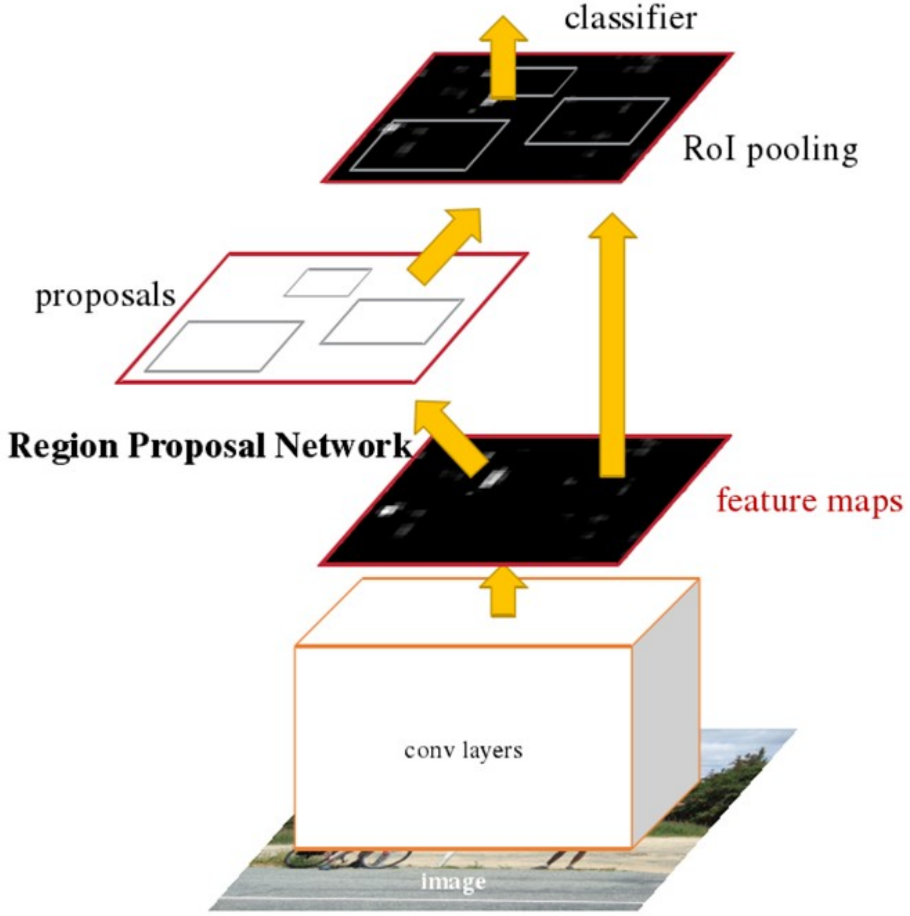
ICCV2019

Wang Xinyu
2022.11.23

https://arxiv.org/pdf/1904.01355.pdf
https://arxiv.org/abs/2006.09214.pdf T-PAMI 2020

Motivation



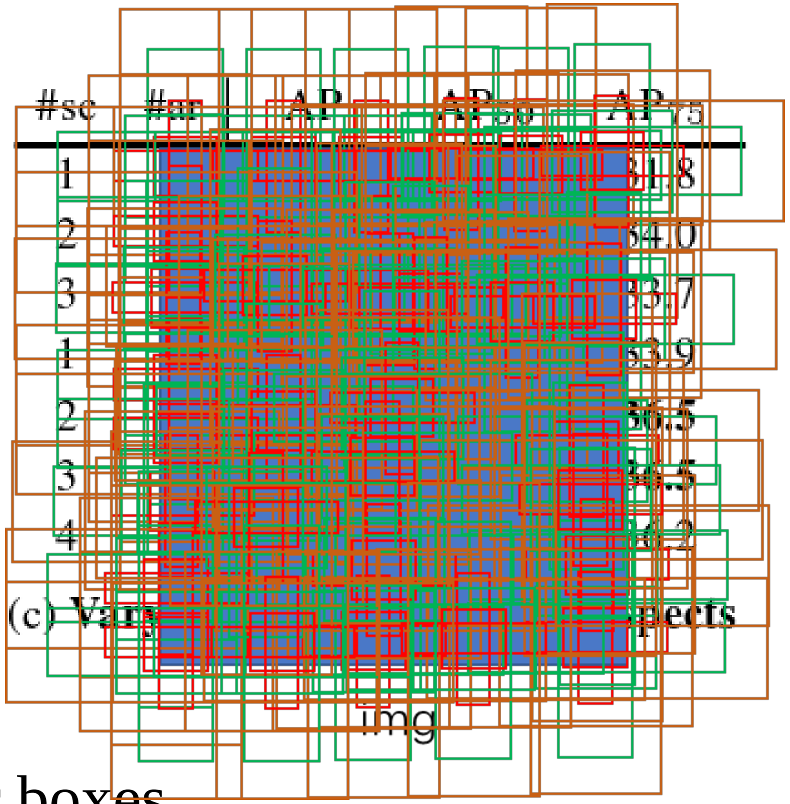


Mainstream detectors:

1、 one-stage： SSD， YOLOv2-v7…

2、 two-stage： Faster R-CNN…

Motivation



- Proposal free and anchor free
- Avoiding the complicated computation
- Improving accuracy with the simpler detector

Drawbacks:

1、 sensitive to the sizes, aspect ratios and number of anchor boxes.

2、 encounter difficulties to deal with object candidates with large shape variations

3、 the imbalance between positive and negative samples in training

4、 complicated computation

FCOS

Method

The feature maps at layer i of a backbone:

$$F_i \in \mathbb{R}^{H \times W \times C}$$

The ground-truth bounding boxes:

$$B_i = (x_0^{(i)}, y_0^{(i)}, x_1^{(i)}, y_1^{(i)}, c^{(i)}) \in \mathbb{R}^4 \times \{1, 2 \dots C\}$$

The regression targets for the location:

$$\boldsymbol{t}^* = (l^*, t^*, r^*, b^*)$$

It can be formulated as,

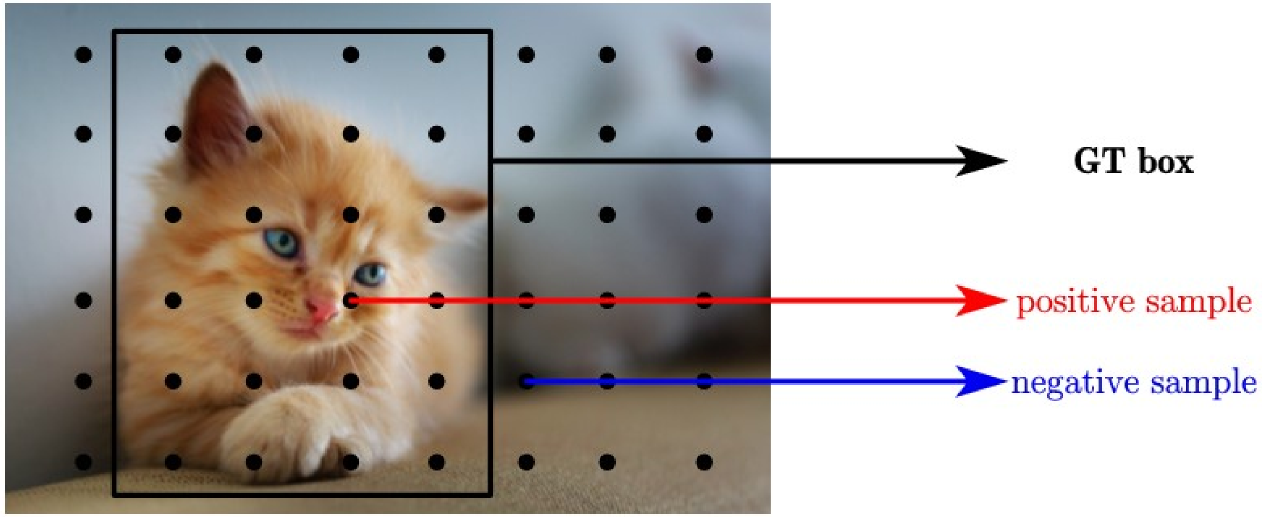
$$l^* = (x - x_0^{(i)})/s, \quad t^* = (y - y_0^{(i)})/s,$$
$$r^* = (x_1^{(i)} - x)/s, \quad b^* = (y_1^{(i)} - y)/s,$$

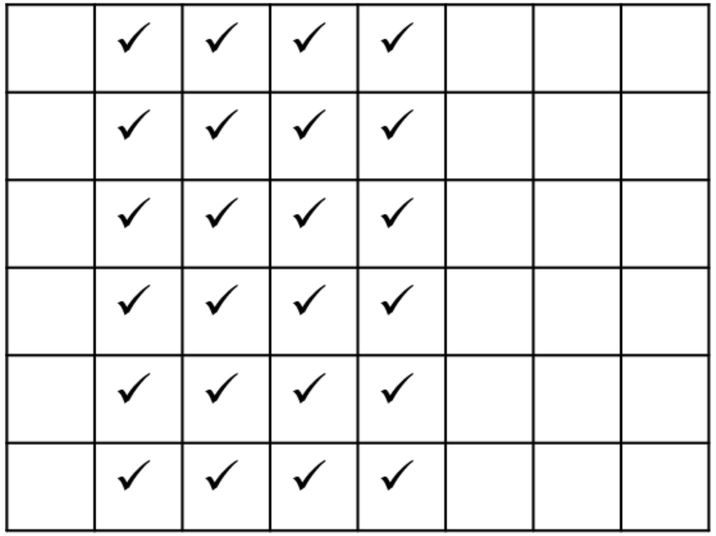
(x,y): *location on the feature map*

$(x_0, y_0), (x_1, y_1)$: *the left top and right bottom corners of the bounding box*

s: *the total stride*

Positive and negative samples for training





feature map

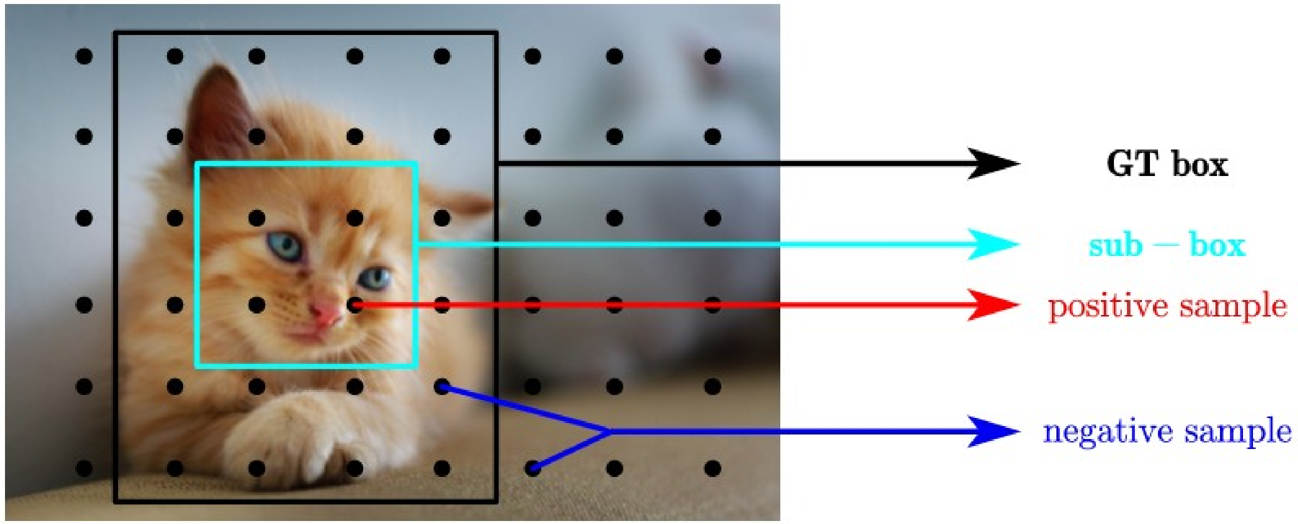
$$(x,\ y) \rightarrow \left(\left\lfloor \frac{s}{2} \right\rfloor + xs,\ \left\lfloor \frac{s}{2} \right\rfloor + ys\right)$$

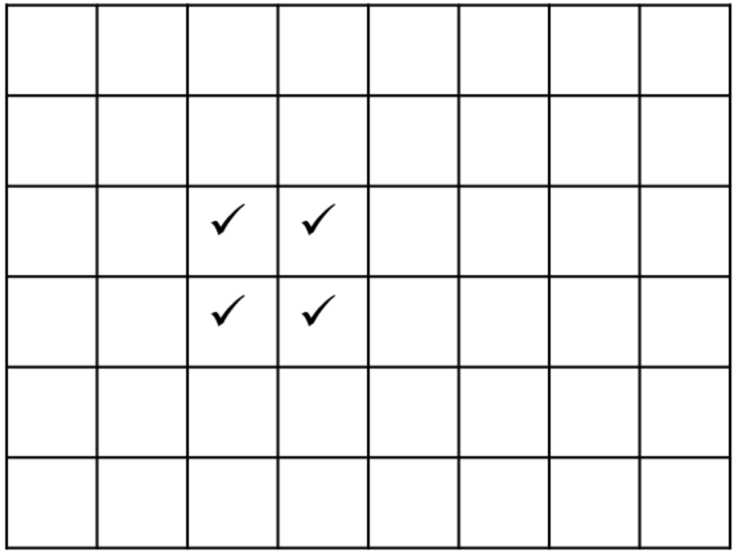
$(x,\ y)$: *location on the feature map* F_i

s: *the total stride until the current feature maps*

location (x, y) is considered as a positive sample if it falls into the **GT box**

Positive and negative samples for training





feature map

$$\textit{\textbf{sub}} - \textit{\textbf{box}} \longrightarrow (c_x - rs, c_y - rs, c_x + rs, c_y + rs,)$$

r: *a hyper − parameter being 1.5 on COCO*

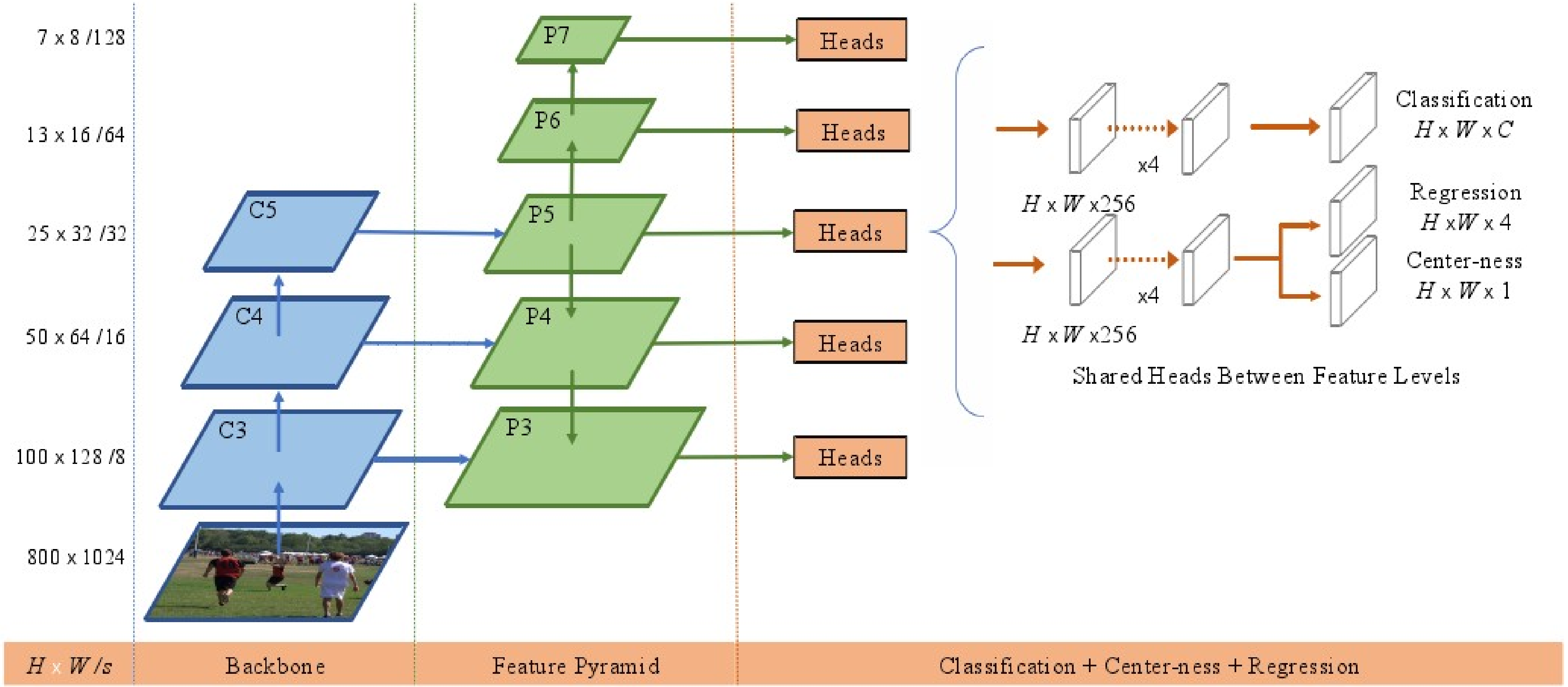
(c_x, c_y): *the center of a ground − truth box*

s: *the total stride until the current feature maps*

location (x, y) is considered as a positive sample if it falls into the **sub-box**

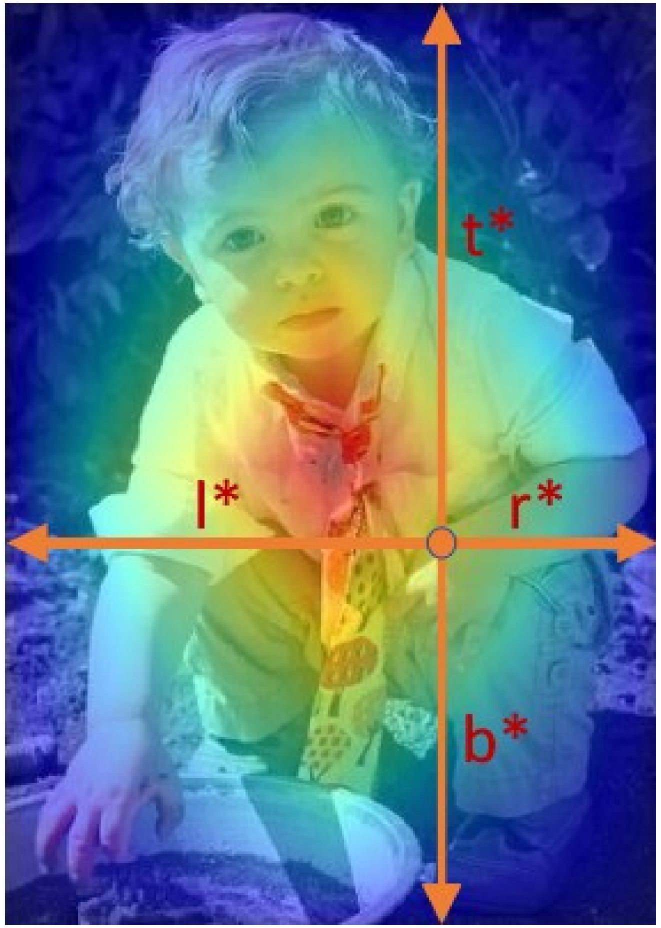
r	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
1.0	38.5	57.2	41.5	22.6	42.3	49.7
1.5	**38.9**	57.5	**42.2**	**23.1**	**42.7**	**50.2**
2.0	38.8	**57.7**	41.7	22.7	42.6	49.9

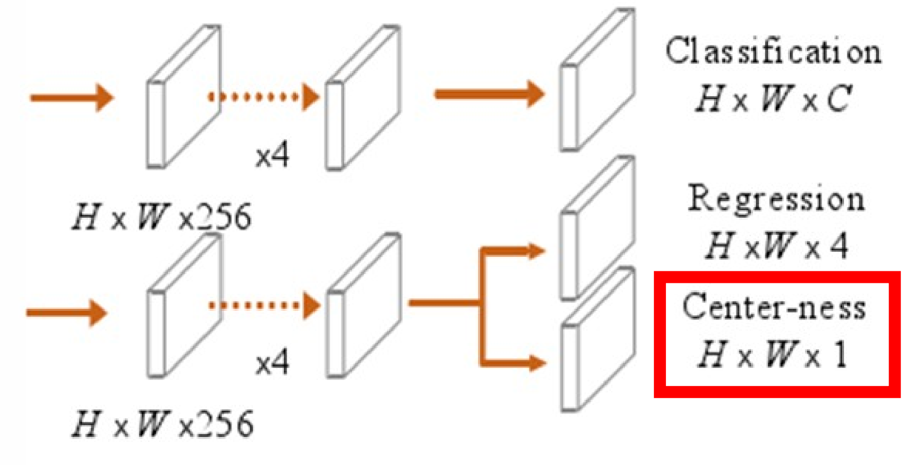
Fully Convolutional One-Stage Object Detector



- **Backbone:** Restnet50
- **FPN:** P6 and P7 are produced by applying one 3 × 3 convolutional layer with the stride being 2 on P5 and P6
- **Head:** Shared heads, Classification branch, Regression branch and Center-ness branch

Center-ness for FCOS





the center-ness target (used for regression) is defined as:

$$\text{centerness}^* = \sqrt{\frac{\min(l^*, r^*)}{\max(l^*, r^*)} \times \frac{\min(t^*, b^*)}{\max(t^*, b^*)}}.$$

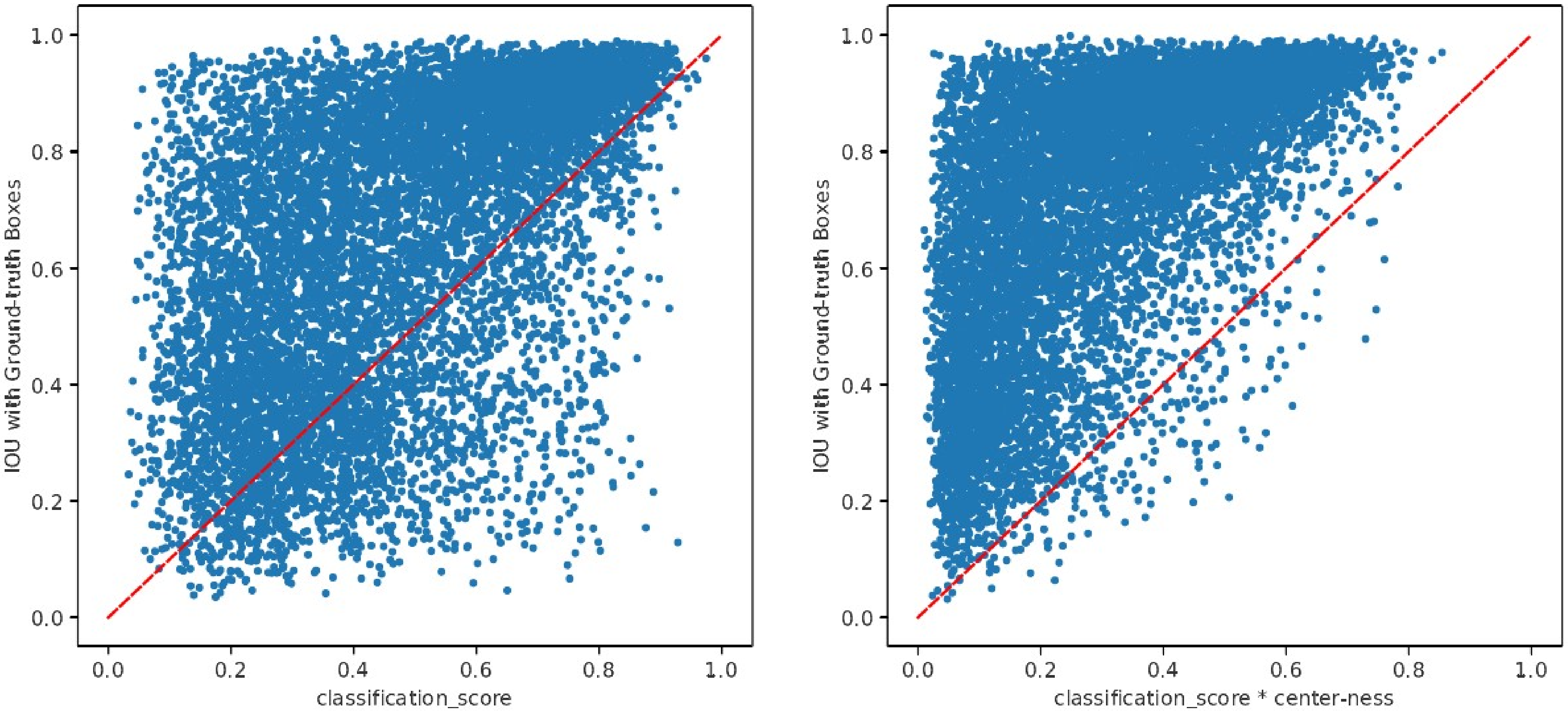
the final score (used for NMS):

$$\boldsymbol{s}_{x,y} = \sqrt{\boldsymbol{p}_{x,y} \times o_{x,y}},$$

:predicted center-ness

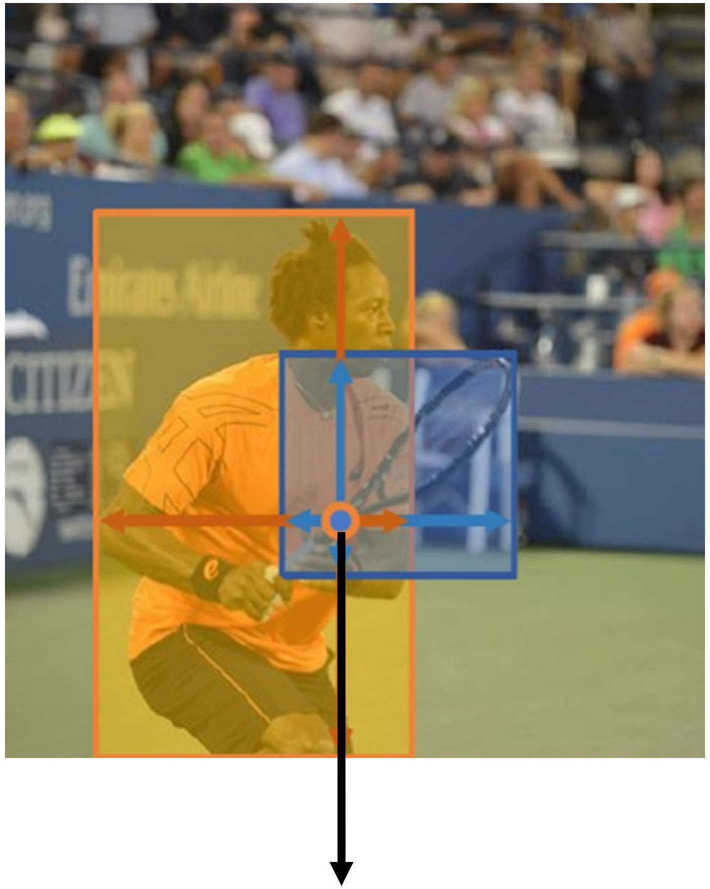
:classification score

Center-ness for FCOS

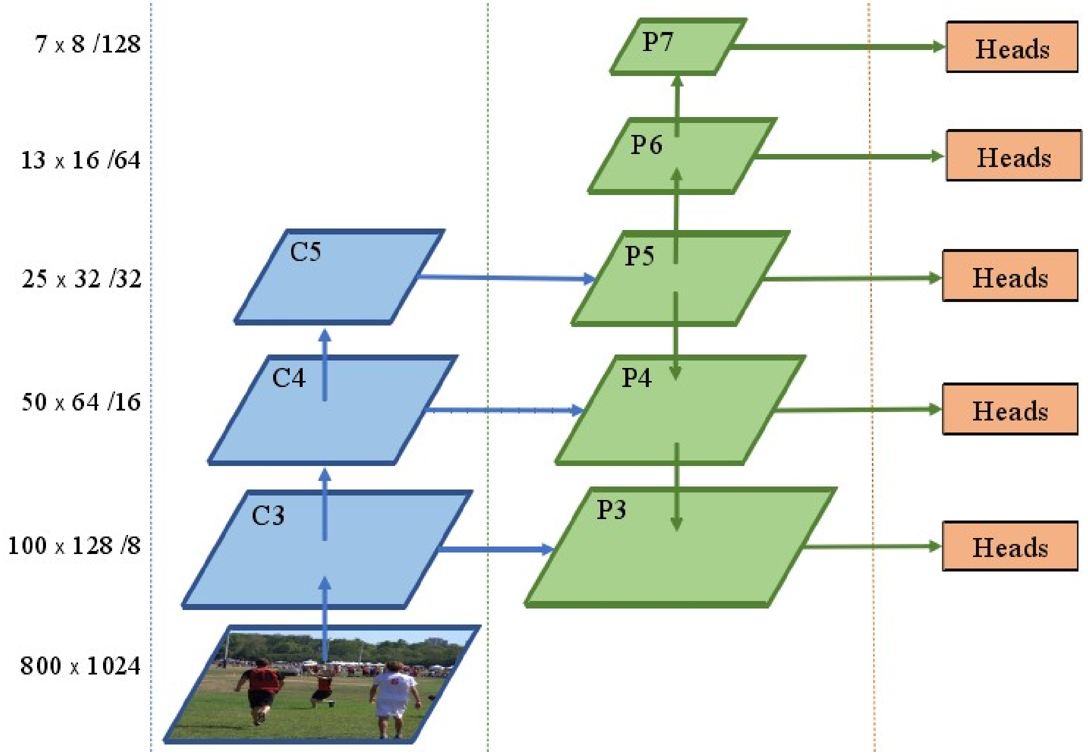


Multi-level Prediction with FPN

Ambiguous Sample Problem



Ambiguous sample



$$Positive\ sample:\ m_{i-1} < \max(l^*, t^*, r^*, b^*) < m_i$$

$$Negative\ sample:\ \max(l^*, t^*, r^*, b^*) > m_i\ or\ \max(l^*, t^*, r^*, b^*) < m_{i-1}$$

$$i = 3, 4, 5, 6, 7$$

$$m_2 = 0\ , m_3 = 64\ , m_4 = 128\ , m_5 = 256\ , m_6 = 512\ , m_7 = \infty$$

Loss Function

$$L(\{p_{x,y}\},\{t_{x,y}\},\{s_{x,y}\}) = \frac{1}{N_{pos}}\sum_{x,y} L_{cls}(p_{x,y}, c^*_{x,y}) + \frac{1}{N_{pos}}\sum_{x,y} 1_{\{c^*_{x,y}>0\}} L_{reg}(t_{x,y}, t^*_{x,y}) + \frac{1}{N_{pos}}\sum_{x,y} 1_{\{c^*_{x,y}>0\}} L_{ctrness}(s_{x,y}, s^*_{x,y})$$

- $p_{x,y}$表示在特征图(x,y)点处预测的每个类别的score
- $c^*_{x,y}$表示在特征图(x,y)点对应的真实类别标签
- $1_{\{c^*_{x,y}>0\}}$当特征图(x,y)点被匹配为正样本时为1，否则为0
- $t_{x,y}$表示在特征图(x,y)点处预测的目标边界框信息
- $t^*_{x,y}$表示在特征图(x,y)点对应的真实目标边界框信息
- $s_{x,y}$表示在特征图(x,y)点处预测的 `center-ness`
- $s^*_{x,y}$表示在特征图(x,y)点对应的真实 `center-ness`

$$\mathrm{FL}(p_t) = -(1-p_t)^\gamma \log(p_t).$$

$Classification\ loss\ L_{cls}$: **Focal Loss**

$Regression\ loss\ L_{reg}$: **GIoU Loss**

$Center-ness\ loss\ L_{ctrness}$: **BCE**

$$GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$$

Experiments

- Backbone: Restnet50
- Pre-trained: ImageNet
- Inference: input images, obtain the classification scores and the regression prediction for each location

1.Ablation Study

1.1 Multi-level Prediction with FPN

a. BPR :Best Possible Recall

Method	w/ FPN	Low-quality matches	BPR (%)
RetinaNet	✓	Not used	88.16
RetinaNet	✓	≥ 0.4	91.94
RetinaNet	✓	All	**99.32**
FCOS		-	96.34
FCOS	✓	-	98.95

TABLE 1
The best possible recall (BPR) of anchor-based RetinaNet under a variety of matching rules and the BPR of FCOS on the COCO `val2017` split. FCOS has very similar BPR to the best anchor-based one and has much higher recall than the official implementation in `Detectron` [46], where only low-quality matches with IOU ≥ 0.4 are considered.

b. Ambiguous Samples

w/ ctr. sampling	w/ FPN	1	2	≥ 3
		76.60%	20.05%	3.35%
	✓	92.58%	6.97%	0.45%
✓		96.52%	3.34%	0.14%
✓	✓	97.34%	2.59%	0.07%

TABLE 2
The ratios of the ambiguous samples to all the positive samples in FCOS. $1, 2$ and ≥ 3 denote the number of ground-truth boxes a location should be associated to. If the number is greater than 1, the location is defined as an "ambiguous sample" in this work. As shown in the table, with center sampling and FPN, the ratio of ambiguous samples is low ($i.e.$, $< 3\%$).

Experiments

1.2 Center-ness

	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
w/o ctr.-ness	38.0	57.2	40.9	21.5	42.4	49.1
w/ ctr.-ness†	37.5	56.5	40.2	21.6	41.5	48.5
w/ ctr.-ness (L1)	**38.9**	**57.6**	42.0	23.0	42.3	**51.0**
w/ ctr.-ness	**38.9**	57.5	**42.2**	**23.1**	**42.7**	50.2

TABLE 4

Ablation study for the proposed center-ness branch on the `val2017` split. ctr.-ness†: using the center-ness computed from the predicted regression vector when testing (*i.e.*, replacing the ground-truth values with the predicted ones in Eq. (3)). "ctr.-ness" is that using center-ness predicted from the proposed center-ness branch. The center-ness branch improves the detection performance. On the contrary, using "ctr.-ness†" even degrades the performance, which suggests that the separate center-ness branch is necessary. w/ ctr.-ness (L1): using L1 instead of BCE as the loss to optimize the center-ness.

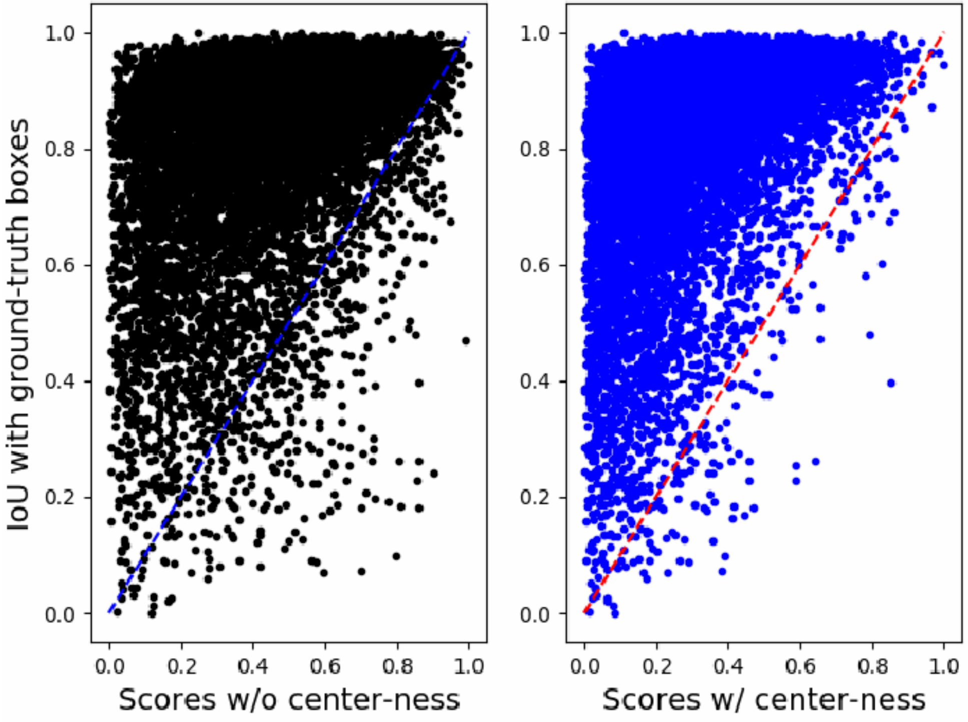


Fig. 5. **Quantitative results of applying the center-ness scores to classification scores**. A point in the figure denotes a bounding box. The dashed line is the line $y = x$. As shown in the right figure, after applying the center-ness scores, the boxes with low IoU scores but high confidence scores (*i.e.*, under the line $y = x$) are reduced substantially.

Experiments

1.3 Other Design Choices

r	AP	AP$_{50}$	AP$_{75}$	AP$_S$	AP$_M$	AP$_L$
1.0	38.5	57.2	41.5	22.6	42.3	49.7
1.5	**38.9**	57.5	**42.2**	**23.1**	**42.7**	**50.2**
2.0	38.8	**57.7**	41.7	22.7	42.6	49.9

TABLE 6
Ablation study for the radius r of positive sample regions (defined in Section 2.1).

	AP	AP$_{50}$	AP$_{75}$	AP$_S$	AP$_M$	AP$_L$
Baseline	**38.9**	**57.5**	**42.2**	**23.1**	**42.7**	**50.2**
w/o GN	37.9	56.4	40.9	22.1	41.8	48.8
w/ IoU	38.6	57.2	41.9	22.4	42.1	49.8
w/ C$_5$	38.5	57.4	41.7	22.8	42.1	49.3

TABLE 5
Ablation study for design choices in FCOS. w/o GN: without using Group Normalization (GN) for the convolutional layers in heads. w/ IoU: using IoU loss in [19] instead of GIoU. w/ C_5: using C_5 instead of P_5.

Strategy	AP	AP$_{50}$	AP$_{75}$	AP$_S$	AP$_M$	AP$_L$
FPN	37.7	56.6	40.6	22.2	40.9	49.7
$\sqrt{(h^* \times w^*)}/2$	37.6	56.5	40.6	22.4	41.6	47.3
$\max(h^*, w^*)/2$	38.1	57.0	41.3	22.5	41.8	48.7
$\max(l^*, t^*, r^*, b^*)$	**38.9**	**57.5**	**42.2**	**23.1**	**42.7**	**50.2**

TABLE 7
Ablation study for different strategies of assigning objects to FPN levels. FPN: the strategy of assigning object proposals (*i.e.*, ROIs) to FPN levels in the original FPN, described in the text. h^* and w^* are the height and width of a ground-truth box, respectively. l^*, t^*, r^* and b^* are the distances from a location to the four boundaries of a ground-truth box. "$\max(l^*, t^*, r^*, b^*)$" (used by FCOS) has the best performance.

- Classification 和 Regression Head 中的 Group Normalization
- GIoU Loss
- 正样本区域参数 r
- FPN 分配策略

Experiments

2. FCOS vs. Anchor-based Counterparts

Method	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	AR_1	AR_{10}	AR_{100}	AR_S	AR_M	AR_L
RetinaNet (#A=9)	35.9	55.8	38.4	20.6	39.8	46.6	31.0	49.8	53.0	33.8	57.4	67.9
RetinaNet (#A=1) w/ imprv.	35.2	55.6	37.0	19.9	39.2	45.2	30.4	49.9	53.5	33.6	57.7	68.2
RetinaNet (#A=9) w/ imprv.	37.6	56.6	40.6	21.5	42.1	48.0	32.1	52.2	56.4	35.5	60.2	72.7
FCOS w/o ctr.-ness	38.0	57.2	40.9	21.5	42.4	49.1	32.1	52.4	56.2	36.6	60.6	71.9
FCOS w/ ctr.-ness	**38.9**	*57.5*	**42.2**	**23.1**	**42.7**	**50.2**	**32.4**	**53.8**	**57.5**	**38.5**	**62.1**	**72.9**

TABLE 3

FCOS vs. RetinaNet on `val2017` split with ResNet-50-FPN as the backbone. All experiments use the same training settings. The proposed anchor-free FCOS achieves even better performance than anchor-based RetinaNet. #A: the number of anchors per location. RetinaNet (#A=9): the original RetinaNet from `Detectron2` [49]. RetinaNet w/ imprv. RetinaNet with the universal improvements in FCOS including Group Normalization (GN) [50], GIoU loss [42] and scalars in regression, using P_5 instead of C_5 and NMS threshold 0.6 instead of 0.5. We have tried our best to make all the details consistent. As shown the table, even without the center-ness branch, the much simpler FCOS already outperforms "RetinaNet (#A=9) w/ imprv" by 0.4% in AP. With the center-ness branch, the performance is further improved to 38.9% in AP.

Method	Backbone	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Two-stage methods:							
Faster R-CNN+++ [47]	ResNet-101	34.9	55.7	37.4	15.6	38.7	50.9
Faster R-CNN w/ FPN [9]	ResNet-101-FPN	36.2	59.1	39.0	18.2	39.0	48.2
Faster R-CNN by G-RMI [51]	Inception-ResNet-v2 [52]	34.7	55.5	36.7	13.5	38.1	52.0
Faster R-CNN w/ TDM [53]	Inception-ResNet-v2-TDM	36.8	57.7	39.2	16.2	39.8	52.1
One-stage methods:							
YOLOv2 [30]	DarkNet-19 [30]	21.6	44.0	19.2	5.0	22.4	35.5
SSD513 [5]	ResNet-101-SSD	31.2	50.4	33.3	10.2	34.5	49.8
YOLOv3 608×608 [6]	Darknet-53	33.0	57.9	34.4	18.3	35.4	41.9
DSSD513 [54]	ResNet-101-DSSD	33.2	53.3	35.2	13.0	35.4	51.1
RetinaNet [7]	ResNet-101-FPN	39.1	59.1	42.3	21.8	42.7	50.2
CornerNet [32]	Hourglass-104	40.5	56.5	43.1	19.4	42.7	53.9
FSAF [33]	ResNeXt-64x4d-101-FPN	42.9	63.8	46.3	26.6	46.2	52.7
CenterNet511 [55]	Hourglass-104	44.9	62.4	48.1	25.6	47.4	57.4
FCOS	ResNet-101-FPN	43.2	62.4	46.8	26.1	46.2	52.8
FCOS	ResNeXt-32x8d-101-FPN	44.1	63.7	47.9	27.4	46.8	53.7
FCOS	ResNeXt-64x4d-101-FPN	44.8	64.4	48.5	27.7	47.4	55.0
FCOS w/ deform. conv. v2 [56]	ResNeXt-32x8d-101-FPN	46.6	65.9	50.8	28.6	49.1	58.6
FCOS	ResNet-101-BiFPN [57]	45.0	63.6	48.7	27.0	47.9	55.9
FCOS	ResNeXt-32x8d-101-BiFPN	46.2	65.2	50.0	28.7	49.1	56.5
FCOS w/ deform. conv. v2	ResNeXt-32x8d-101-BiFPN	47.9	66.9	51.9	30.2	50.3	59.9
w/ test-time augmentation:							
FCOS	ResNet-101-FPN	45.9	64.5	50.4	29.4	48.3	56.1
FCOS	ResNeXt-32x8d-101-FPN	47.0	66.0	51.6	30.7	49.4	57.1
FCOS	ResNeXt-64x4d-101-FPN	47.5	66.4	51.9	31.4	49.7	58.2
FCOS w/ deform. conv. v2	ResNeXt-32x8d-101-FPN	49.1	68.0	53.9	31.7	51.6	61.0
FCOS	ResNet-101-BiFPN	47.9	65.9	52.5	31.0	50.7	59.7
FCOS	ResNeXt-32x8d-101-BiFPN	49.0	67.4	53.6	32.0	51.7	60.5
FCOS w/ deform. conv. v2	ResNeXt-32x8d-101-BiFPN	**50.4**	**68.9**	**55.0**	**33.2**	**53.0**	**62.7**

TABLE 8

FCOS vs. other state-of-the-art two-stage or one-stage detectors (*single-model results*). FCOS outperforms a few recent anchor-based and anchor-free detectors by a considerable margin.

ObjectBox: From Centers to Boxes for Anchor-Free Object Detection

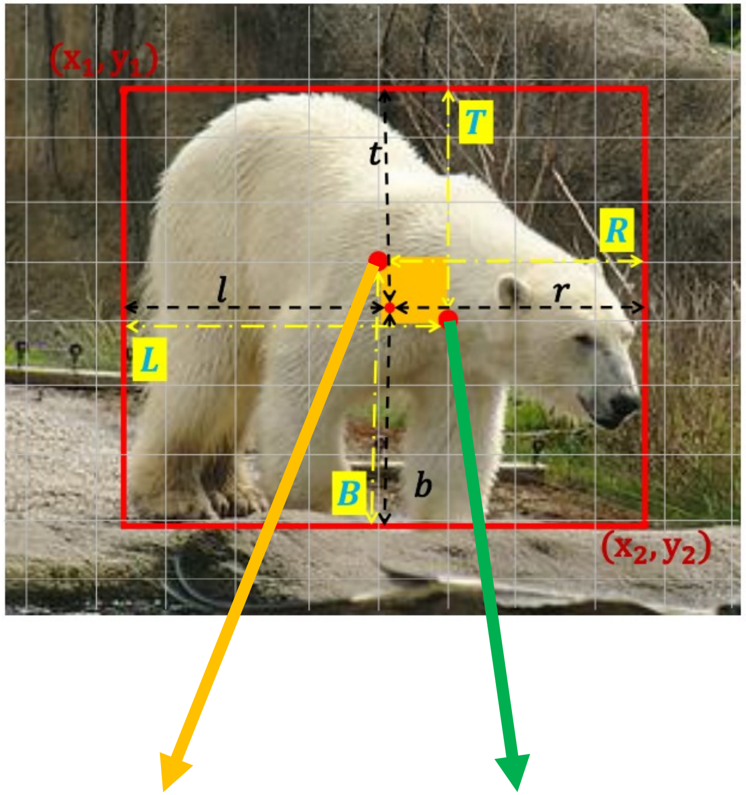
Mohsen Zand , Ali Etemad , and Michael Greenspan

Dept. of Electrical and Computer Engineering, Ingenuity Labs Research InstituteQueen's University, Kingston, Ontario, Canada

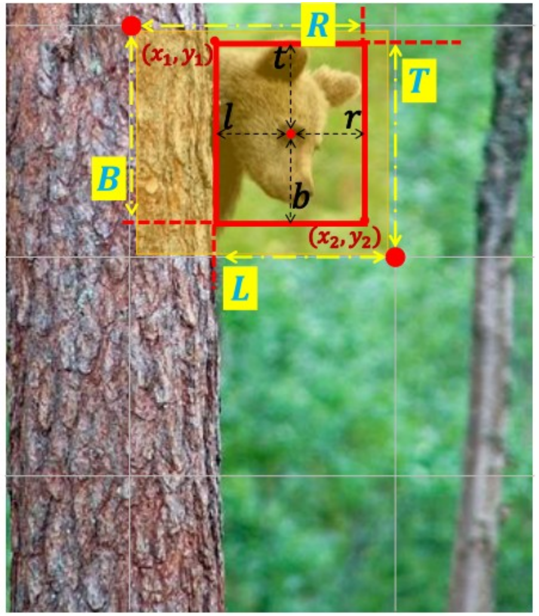
ECCV 2022 Oral

S. Zhang, C. Chi, Y. Yao, Z. Lei and S. Z. Li, "Bridging the Gap Between Anchor-Based and Anchor-Free Detection via Adaptive Training Sample Selection," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 9756-9765, doi: 10.1109/CVPR42600.2020.00978. CVPR 2020

Method



$$(\lfloor \frac{x}{s_i} \rfloor, \lfloor \frac{y}{s_i} \rfloor) \quad (\lfloor \frac{x}{s_i} \rfloor + 1, \lfloor \frac{y}{s_i} \rfloor + 1)$$



The regression targets at scale i:

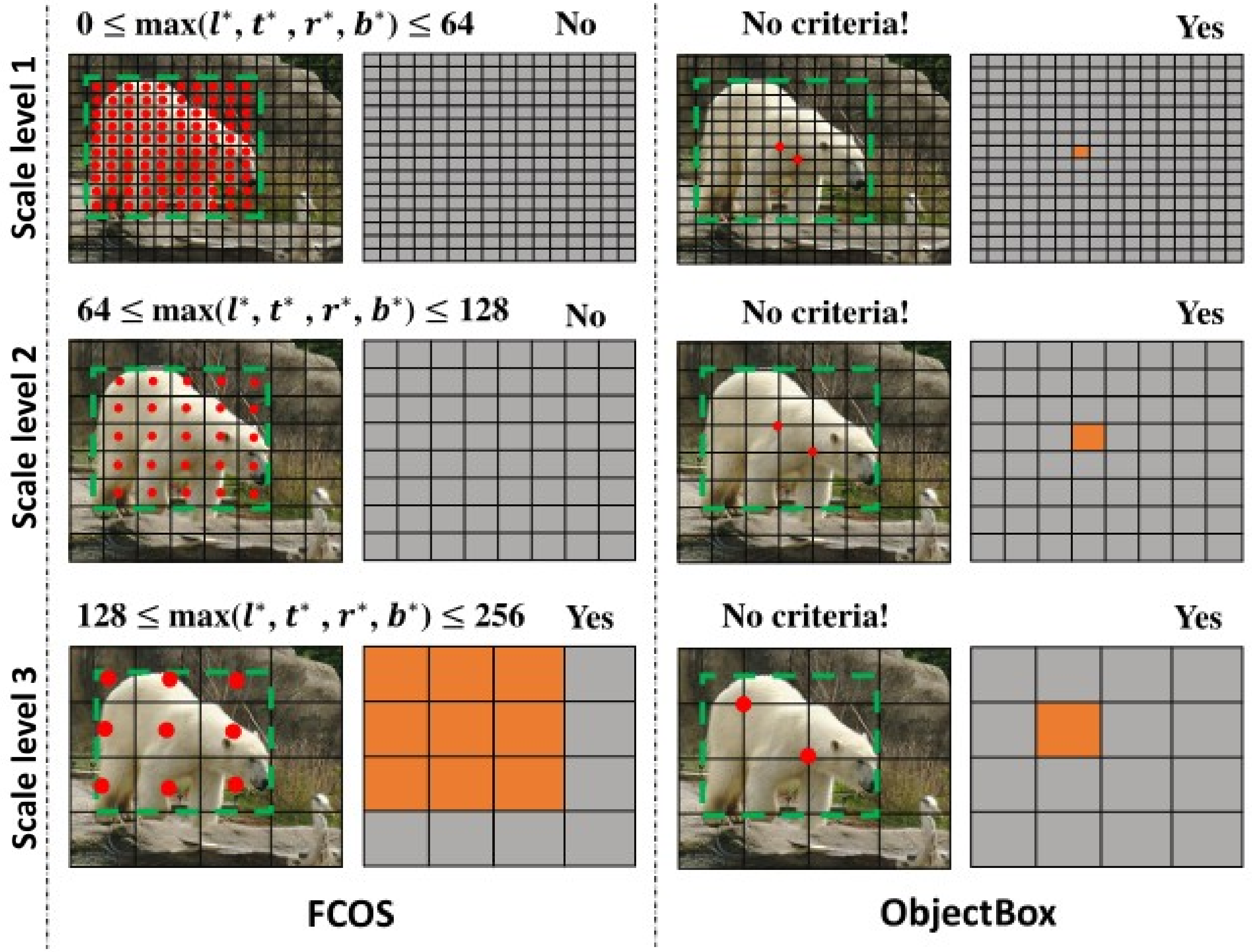
$$\begin{cases} L^{(i)^*} = (\lfloor \frac{x}{s_i} \rfloor + 1) - (x_1^{(i)}/s_i) \\ T^{(i)^*} = (\lfloor \frac{y}{s_i} \rfloor + 1) - (y_1^{(i)}/s_i) \\ R^{(i)^*} = (x_2^{(i)}/s_i) - \lfloor \frac{x}{s_i} \rfloor \\ B^{(i)^*} = (y_2^{(i)}/s_i) - \lfloor \frac{y}{s_i} \rfloor \end{cases}$$

(x,y): *the center of the bounding box*

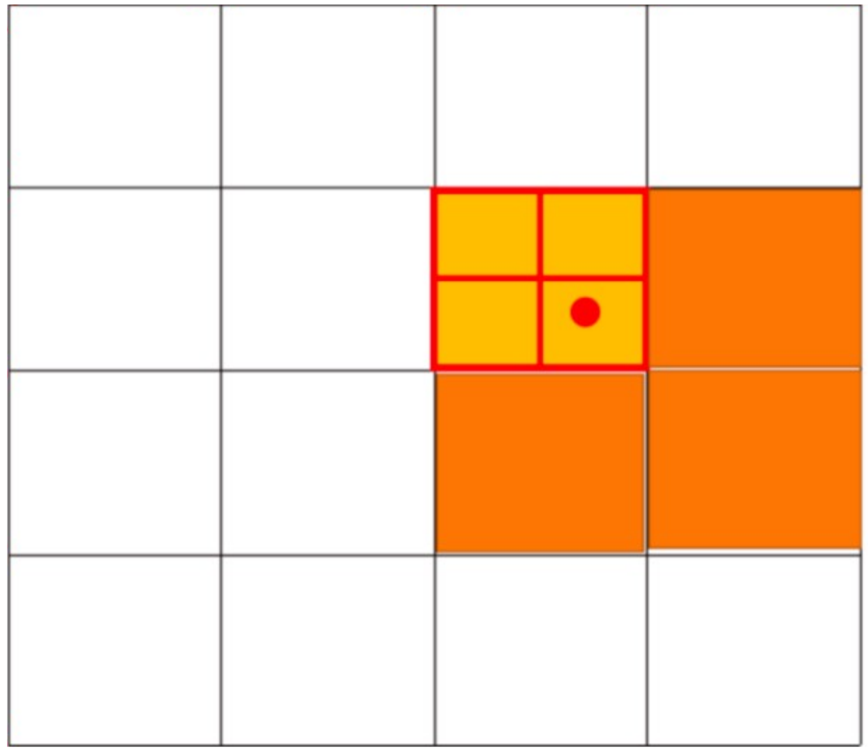
$(x_1,y_1),(x_2,y_2)$: *the left top and right bottom corners of the bounding box*

s: *the total stride*

Label assignment

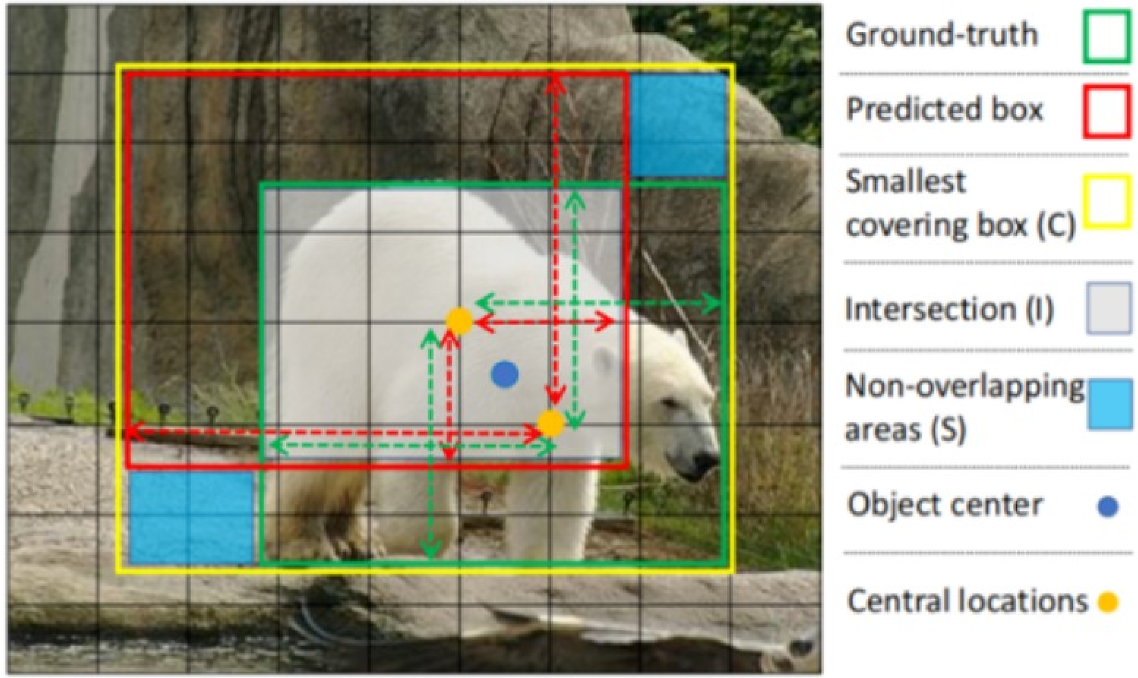


The center of the bounding box



Loss Function

$$\ell^s = \ell^s_{cls} + \ell^s_{obj} + \boxed{\ell^s_{box}}$$



Method	Avg. Precision, IoU			Avg. Precision, Area		
	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
MSE	22.6	44.1	19.4	12.5	18.3	35.7
Adopted GIoU	27.4	46.9	28.2	23.8	30.2	41.8
Adopted CIoU	27.1	46.5	28.1	24.0	30.5	41.0
SDIoU	46.8	65.9	49.5	26.8	49.5	57.6

The non-overlapping area S :

$$S = (L^* - L)^2 + (T^* - T)^2 + (R^* - R)^2 + (B^* - B)^2$$

The intersection area I :

$$I = (w^I)^2 + (h^I)^2$$

$$w^I = min(L^*, L) + min(R^*, R) - 1$$
$$h^I = min(T^*, T) + min(B^*, B) - 1.$$

The smallest area that covers both predicted and ground-truth boxes C :

$$C = (w^C)^2 + (h^C)^2$$

$$w^C = max(L^*, L) + max(R^*, R) - 1$$
$$h^C = max(T^*, T) + max(B^*, B) - 1.$$

Experiments

Table 1. Performance comparison with the state-of-the-art methods on the MS-COCO dataset in single-model and single-scale results. The bold and underlined numbers respectively indicate the best and second best results in each column

Method	Backbone	Avg. Precision, IoU			Avg. Precision, Area			Avg. Recall, # Dets			Avg. Recall, Area		
		AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	AR_1	AR_{10}	AR_{100}	AR_S	AR_M	AR_L
SSD513 [20]	ResNet-101	31.2	50.4	33.3	10.2	34.5	49.8	28.3	42.1	44.4	17.6	49.2	65.8
DeNet [30]	ResNet-101	33.8	53.4	36.1	12.3	36.1	50.8	29.6	42.6	43.5	19.2	46.9	64.3
F-RCNN w/ FPN [17]	ResNet-101	36.2	59.1	39.0	18.2	39.0	48.2	-	-	-	-	-	-
YOLOv2 [23]	DarkNet-19	21.6	44.0	19.2	5.0	22.4	35.5	20.7	31.6	33.3	9.8	36.5	54.4
RetinaNet [18]	ResNet-101	39.1	59.1	42.3	21.8	42.7	50.2	-	-	-	-	-	-
YOLOv3 [24]	DarkNet-53	33.0	57.9	34.4	18.3	35.4	41.9	-	-	-	-	-	-
CornerNet [16]	Hourglass-104	40.6	56.4	43.2	19.1	42.8	54.3	35.3	54.7	59.4	37.4	62.4	**77.2**
CenterNet [4]	Hourglass-52	41.6	59.4	44.2	22.5	43.1	54.1	34.8	55.7	60.1	38.6	63.3	76.9
ExtremeNet [41]	Hourglass-104	40.2	55.5	43.2	20.4	43.2	53.1	-	-	-	-	-	-
FCOS [29]	ResNeXt-101	42.1	62.1	45.2	25.6	44.9	52.0	-	-	-	-	-	-
ASSD513 [34]	ResNet101	34.5	55.5	36.6	15.4	39.2	51.0	29.9	45.6	47.6	22.8	52.2	67.9
SaccadeNet [15]	DLA-34-DCN	40.4	57.6	43.5	20.4	43.8	52.8	-	-	-	-	-	-
YOLOv4 [1]	CSPDarknet	43.5	65.7	47.3	26.7	46.7	53.3	-	-	-	-	-	-
FoveaBox [14]	ResNeXt-101	43.9	63.5	47.7	26.8	46.9	55.6	-	-	-	-	-	-
RetinaNet+CBAF [28]	ResNet-101	43.0	63.2	46.3	25.9	45.6	51.4	-	-	-	-	-	-
ATSS [37]	ResNet-101	43.6	62.1	47.4	26.1	47.0	53.6	-	-	-	-	-	-
PAA [13]	ResNet-101	44.8	63.3	48.7	26.5	48.8	56.3	-	-	-	-	-	-
OTA [6]	ResNet-101	45.3	63.5	49.3	26.9	48.8	56.1	-	-	-	-	-	-
VarifocalNet [36]	ResNet-101	46.0	64.2	**50.0**	**27.5**	49.4	56.9	-	-	-	-	-	-
ObjectBox	ResNet-101	46.1	65.0	48.3	26.0	48.7	57.3	35.3	57.1	60.5	39.2	65.0	76.9
ObjectBox	CSPDarknet	**46.8**	**65.9**	49.5	26.8	**49.5**	**57.6**	**36.0**	**57.5**	**60.7**	**39.4**	**65.2**	77.0

Table S.4. Inference speed comparison

Method	Backbone	# params	FPS	AP
SSD513 [S.4]	ResNet-101	57 M	43	31.2
Faster R-CNN w/ FPN [S.2]	ResNet-101	42 M	26	36.2
YOLOv3 [S.7]	DarkNet-53	65 M	20	33.0
FCOS [S.9]	ResNeXt-101	32 M	50	42.1
ATSS [S.11]	ResNet-101	32 M	50	43.6
ObjectBox	ResNet-101	30 M	70	46.1
ObjectBox	CSPDarknet	86 M	**120**	**46.8**

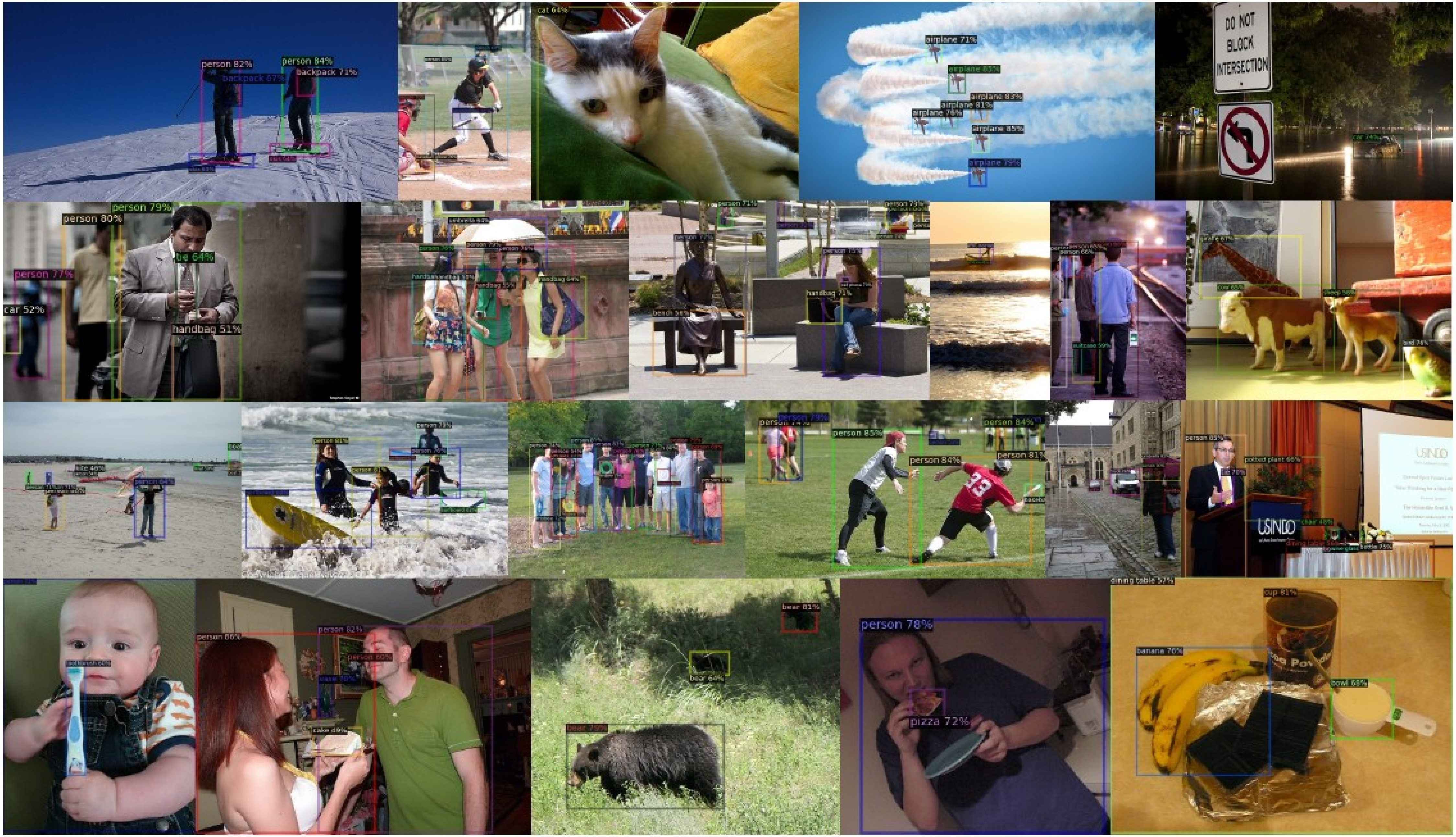
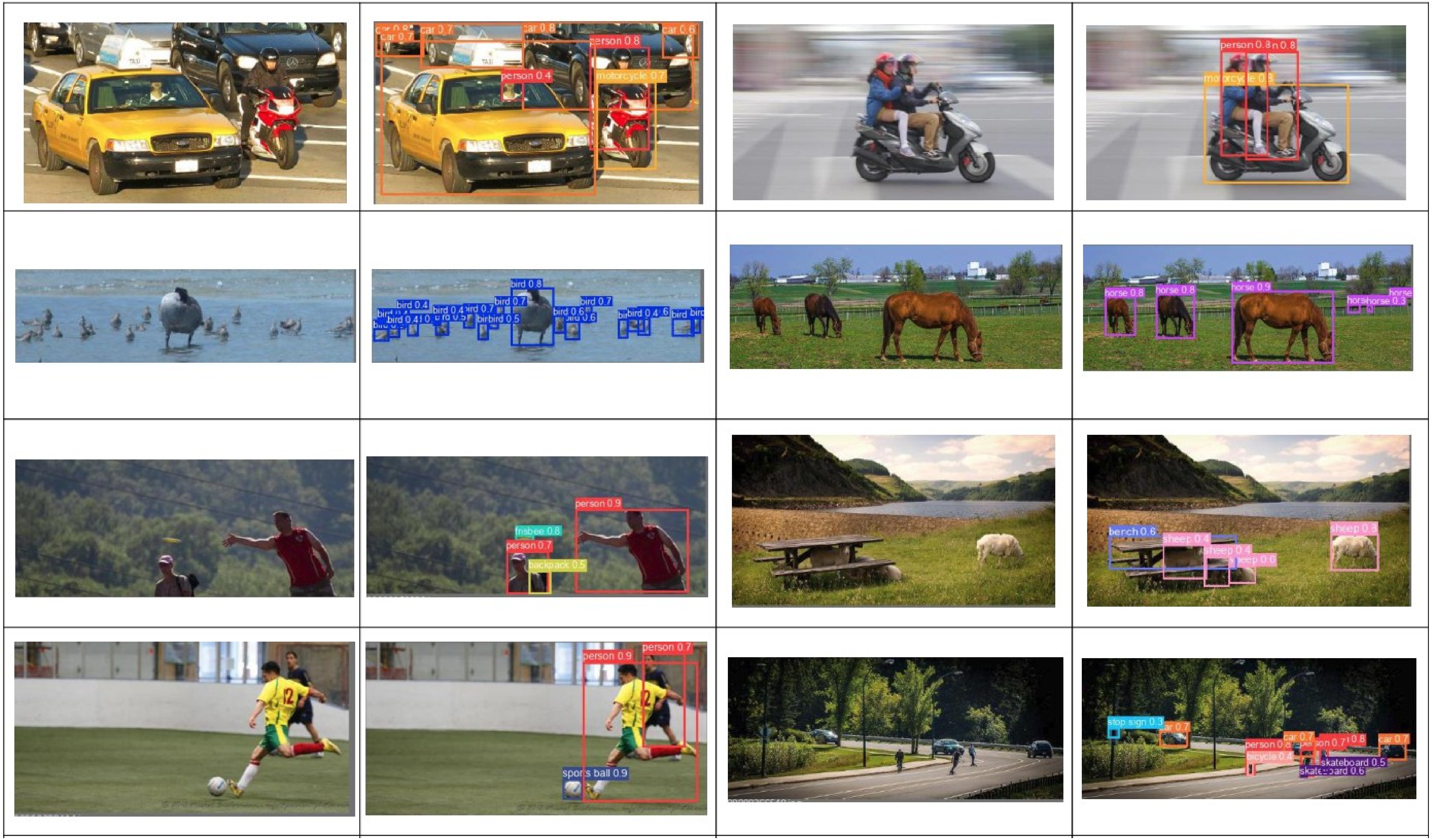
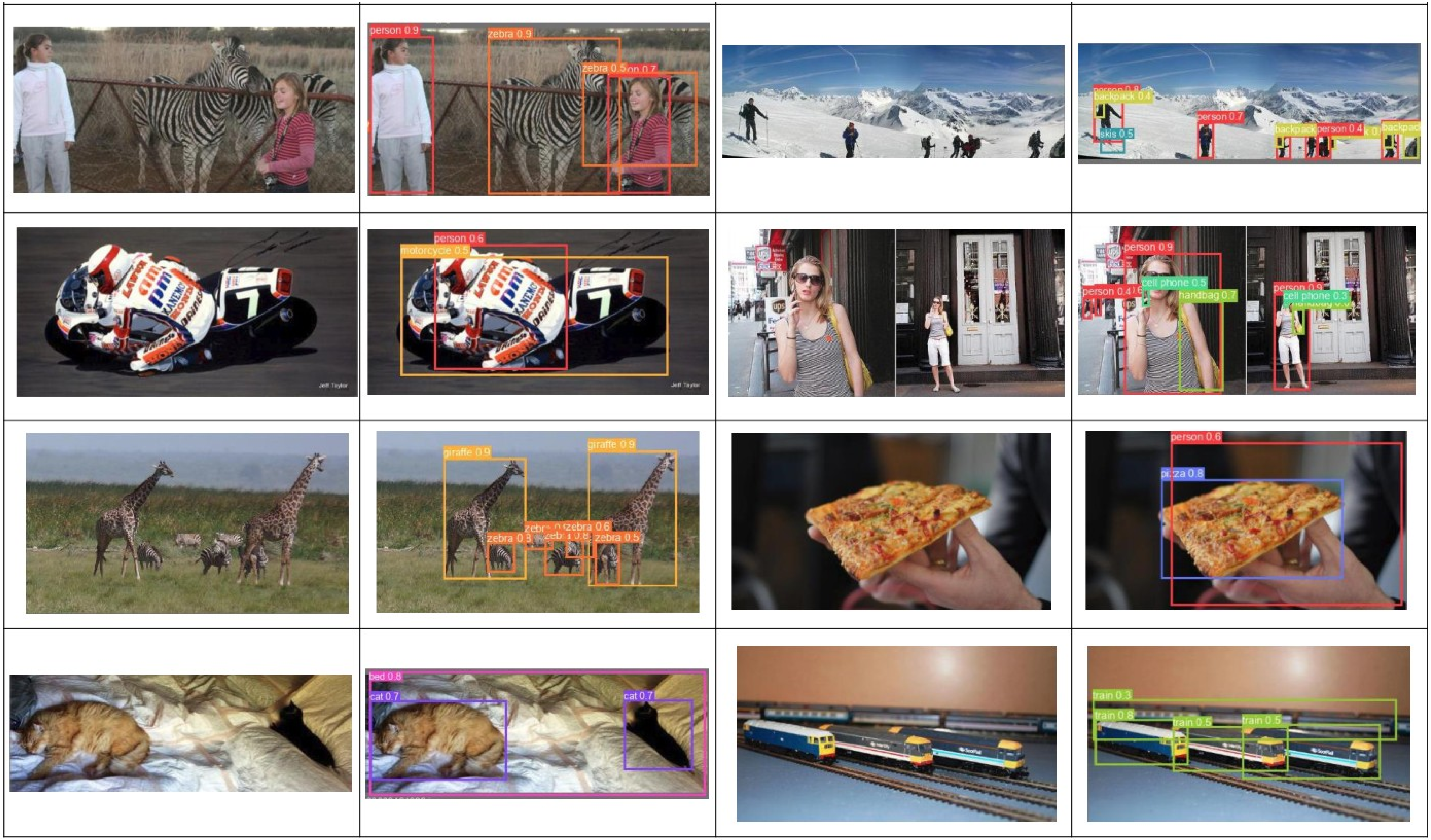


Fig. 6. Qualitative results. As shown in the figure, FCOS works well with a wide range of objects including crowded, occluded, extremely small and very large objects.

Fig. 7. Qualitative results on the CrowdHuman `val` set with the ResNet-50-FPN backbone.


car 0.8
car 0.7
car 0.7
car 0.8
car 0.6
person 0.8
person 0.4
motorcycle 0.7
person 0.8
bird 0.8
bird 0.7
bird 0.4
bird 0.4
bird 0.7
bird 0.5
bird 0.6
bird 0.7
bird 0.6
horse 0.8
horse 0.8
horse 0.9
horse 0.3
person 0.9
frisbee 0.8
person 0.7
backpack 0.5
bench 0.6
sheep 0.4
sheep 0.4
sheep 0.8
person 0.9
person 0.7
sports ball 0.9
stop sign 0.3
car 0.7
car 0.7
bicycle 0.4
skateboard 0.5


person 0.9
zebra 0.9
zebra 0.5
backpack 0.4
person 0.7
skis 0.5
backpack
person 0.4
person 0.6
motorcycle 0.5
person 0.9
cell phone 0.5
person 0.4
handbag 0.7
person 0.9
cell phone 0.3
giraffe 0.9
giraffe 0.9
zebra 0.6
zebra 0.5
person 0.6
pizza 0.8
bed 0.8
cat 0.7
cat 0.7
train 0.3
train 0.8
train 0.5
train 0.5

The development of Anchor-Free detection

1、 Densebox: Unifying landmark localization with end to end object detection. (CVPR2015)
https://arxiv.org/abs/1509.04874

2、 You only look once: Unified, real-time object detection. (CVPR2016)
https://arxiv.org/abs/1506.02640

3、 InUnitbox: An advanced object detection network. (ACM2016)
https://arxiv.org/abs/1608.01471

4、 Cornernet: Detecting objects as paired keypoints. (ECCV2018)
https://arxiv.org/abs/1808.01244v2

5、 FCOS: Fully convolutional one-stage object detection. (ICCV2019)
https://arxiv.org/abs/1904.01355

6、 Objects as points. (CVPR2019)
https://arxiv.org/abs/1904.07850

7、 Bottom-up object detection by grouping extreme and center points. (CVPR2019)
https://arxiv.org/pdf/1901.08043

8、 ObjectBox: From Centers to Boxes for Anchor-Free Object Detection (ECCV2022)
https://arxiv.org/abs/2207.06985

THANKS!