

Anchor-free Vs anchor-based

- RetinaNet、RepPoints、ATSS对比

RetinaNet、RepPoints、ATSS对比

- 即代表了anchor-based、anchor-free、从based到free的弥补手段和各种消融对比实验
- 对比表格链接：
- [论文对比总结\RetinaNet、ATSS、RepPoints对比分析.html](#)

- Adaptive Training Sample Selection (ATSS)

Adaptive Training Sample Selection (ATSS)

- Bridging the Gap Between Anchor-based and Anchor-free Detection via Adaptive Training Sample Selection CVPR.2019
- 通过消融实验发现anchor-free(ATSS)的比anchor-based(Retinanet)的性能提升的根本原因是正负样本选择策略不同(RetinaNet通过anchor box与gt的IOU定义正负样本，FCOS通过location的中心是否在gt中心定义正负样本)。
- 并由此提出自适应样本选择算法(ATSS)，根据目标的统计特征自动选择正负样本。

传统方法的不足

- 因为对anchor设置正负标签与IOU阈值有关。
- 即使anchor的中心点在目标内部，但是如果anchor设置的不好，也不容易得到比较好的正样本。
- 因为高质量的anchor需要高IoU阈值，即 anchor设置得好，就需要高一点的iou去筛选样本； anchor初始状态不佳，就需要低iou去筛选样本，所以需要动态调整IoU阈值，ATSS主要就做了这个事情。

实验1

- 数据集: coco
- RetinaNet加上FCOS的tricks, 可以将RetinaNet从AP 32.5%提升至37.0%, vs (FCOS) 37.8%。说明剩下的0.8%的差距则来自于分类和回归。

Inconsistency	FCOS	RetinaNet (#A=1)					
GroupNorm	✓	✓	✓	✓	✓	✓	✓
GIoU Loss	✓		✓	✓	✓	✓	✓
In GT Box	✓			✓	✓	✓	✓
Centerness	✓				✓	✓	✓
Scalar	✓						✓
AP (%)	37.8	32.5	33.4	34.9	35.3	36.8	37.0

实验2

- 1. 先设定RetinaNet的anchor个数为1，然后将RetinaNet改成了通过FCOS的正负例定义方式，而FCOS改成了通过IOU的正负例定义方式。
- 2. 看表格发现RetinaNet性能提升到37.8%，发现FOCS性能从37.8%降为36.9%。说明了定义方式是如此重要。
- 3. 横着看表格，性能没有多大改变，可见回归方式没有太重要。

Classification \ Regression	Box	Point
	Intersection over Union	36.9
Spatial and Scale Constraint	37.0	37.8
	37.8	37.8

启发性或普适性

- 1. 设置恰当的IoU阈值，从而能筛选出高质量有效的正样本，对于后面的学习非常有帮助。
- 2. 通过消融实验证明增加anchors的数目对性能的提升是没有用的。

Method	#sc	#ar	AP	AP ₅₀	AP ₇₅	AP _S	AP _M	AP _L
RetinaNet (#A=9)	3	3	36.3	55.2	38.8	19.8	39.8	48.8
+Imprs.	3	3	38.4	56.2	41.6	22.2	42.4	50.1
+Imprs.+ATSS	3	3	39.2	57.6	42.7	23.8	42.8	50.9
+Imprs.+ATSS	3	1	39.3	57.7	42.6	23.8	43.5	51.2
+Imprs.+ATSS	1	3	39.2	57.1	42.5	23.2	43.1	50.3
+Imprs.+ATSS	1	1	39.3	57.5	42.8	24.3	43.3	51.3

总结

- ATSS=自适应样本选择(计算所有的IOU的mean (mg)和std dev (vg), 根据自适应阈值 $t_g = mg + vg$, 并增加条件: center在物体内部, 对anchors进行筛选)
- 非常好的对比实验+行文思路清晰

我的观点

- 1.指出了anchor-based检测与anchor-free检测的本质区别。
 - 2.提出了一种自适应训练样本选择(ATSS)，根据目标的统计特征自动选择正负样本。
- 3.讨论了在图像上每个位置平铺多个anchor点来检测目标的必要性。

- RepPoints: Point Set Representation for Object Detection CVPR.2019

RepPoints

- RepPoints: 替代边界框， RepPoints: 替代边界框，基于点集的更细粒度的目标表示新方法RepPoints。
- 通过一组点集提供更细粒度的位置表示和便于分类的信息。
- 作者不使用bbox，使用object周围的9个key points用于更好的定位和更好的对象特征提取。

传统方法的不足

- 4-d的bounding box是目标位置的粗糙表达，边界框只考虑目标的矩形空间范围，不考虑形状、姿态和语义上重要的局部区域的位置(导致当proposal和GT之间所需的细化很小时，它在实践中表现得很好，但它们之间的距离较大时，效果则不好。)
- 而本文的RepPoints是物体一组自适应的特征点集，反映了更精确的语义定位，可用于更好的定位和更好的对象特征提取。



实验

- 1.与anchor-based相比，RepPoints使用 ResNet-50 作为主干网络时提升 2.1 mAP，使用 ResNet-101 时提升了 2.0 mAP。
- 2. 没有 multi-scale 训练和测试的情况下，使用 ResNet-101-DCN 主干网络实现了 42.8 AP，性能优于现有的所有不采用 anchor 的检测器。此外，RPDet 获得了 65.0 的 AP_{50} ，大大超过了所有基线。
- 3. 在train和test都采用multi-scale策略，RPDet可达到46.5mAP。

实验结果

	Backbone	Anchor-Free	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Faster R-CNN w. FPN [24]	ResNet-101		36.2	59.1	39.0	18.2	39.0	48.2
RefineDet [46]	ResNet-101		36.4	57.5	39.5	16.6	39.9	51.4
RetinaNet [25]	ResNet-101		39.1	59.1	42.3	21.8	42.7	50.2
Deep Regionlets [44]	ResNet-101		39.3	59.8	-	21.7	43.7	50.9
Mask R-CNN [13]	ResNeXt-101		39.8	62.3	43.4	22.1	43.2	51.2
FSAF [50]	ResNet-101		40.9	61.5	44.0	24.0	44.2	51.3
Cascade R-CNN [2]	ResNet-101		42.8	62.1	46.3	23.7	45.5	55.2
CornerNet [21]	Hourglass-104	✓	40.5	56.5	43.1	19.4	42.7	53.9
ExtremeNet [48]	Hourglass-104	✓	40.1	55.3	43.2	20.3	43.2	53.1
RPDet	ResNet-101	✓	41.0	62.9	44.3	23.6	44.1	51.7
RPDet	ResNet-101-DCN	✓	42.8	65.0	46.3	24.9	46.2	54.7
RPDet (ms train)	ResNet-101-DCN	✓	45.0	66.1	49.0	26.6	48.6	57.5
RPDet (ms train & ms test)	ResNet-101-DCN	✓	46.5	67.4	50.9	30.3	49.7	57.1

普适性或启发性

- 1.学习像 RepPoints 这样更丰富、更自然的物体表示方法反映了更精确的语义定位，在目标的定位和对象提取上更有优势。

总结

- RepPoints = RepPoints + RepPoints regression + RPDet

我的观点

RepPoints通过引入key points的方法有利于更好提取更精确的语义信息，RepPoints最出色的地方在于增加了对于可形变卷积的监督。

- 因为可形变卷积有很强的表达能力，很好的性能，因此尝试加强对于可形变卷积的监督，可增加可形变卷积的特征点和目标检测中物体的联系。

Representation	Supervision		AP	AP_{50}	AP_{75}
	loc.	rec.			
bounding box	✓		36.2	57.3	39.8
	✓	✓	36.2	57.5	39.8
RepPoints		✓	33.8	54.3	35.8
	✓		37.6	59.4	40.4
	✓	✓	38.3	60.0	41.1

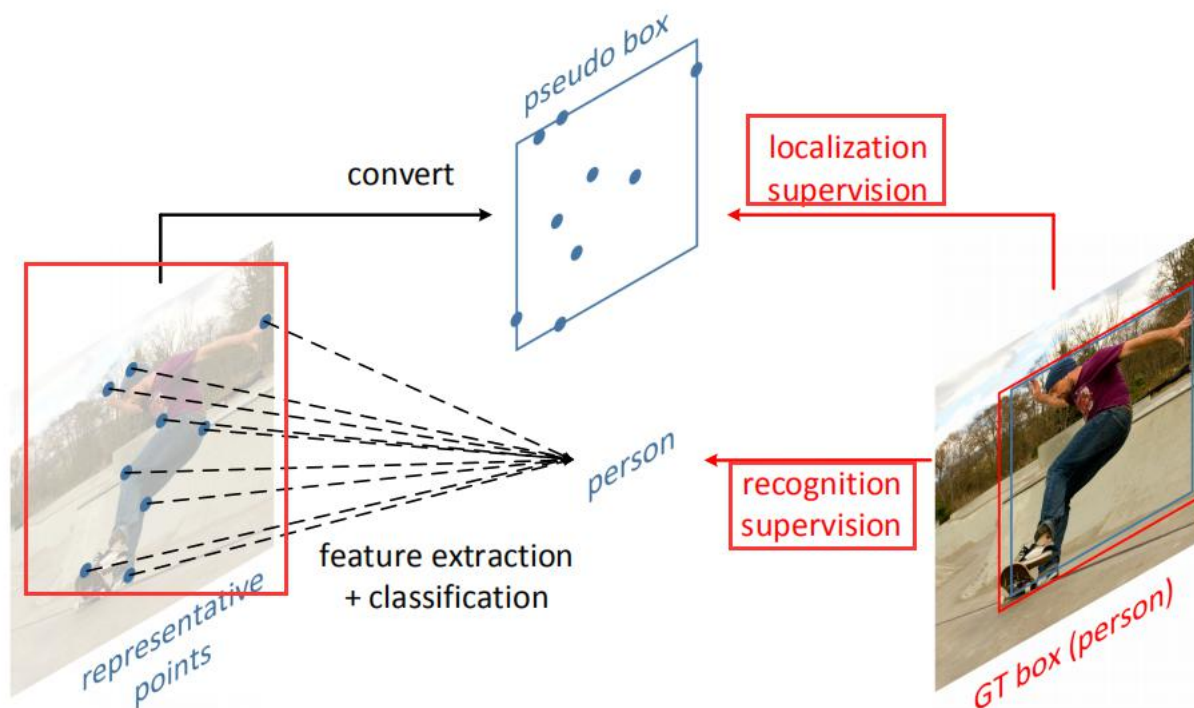
Outline

存在问题

- 1. RepPoints点的初始化是如何被初始化的？
- 2. SNIP、TridentNet中的valid the area of an RoI range与ATSS的做法有点相似，这里前者是对三个scale设置valid IoU range，后者是根据统计特征一个是自适应IoU阈值($tg = mg + vg$)，不知道这两种策略的单独对比的实验效果。

RepPoints

RepPoints点的初始化是如何被初始化的？



To overcome the above limitations, *RepPoints* instead models a set of adaptive sample points:

$$\mathcal{R} = \{(x_k, y_k)\}_{k=1}^n, \quad (4)$$

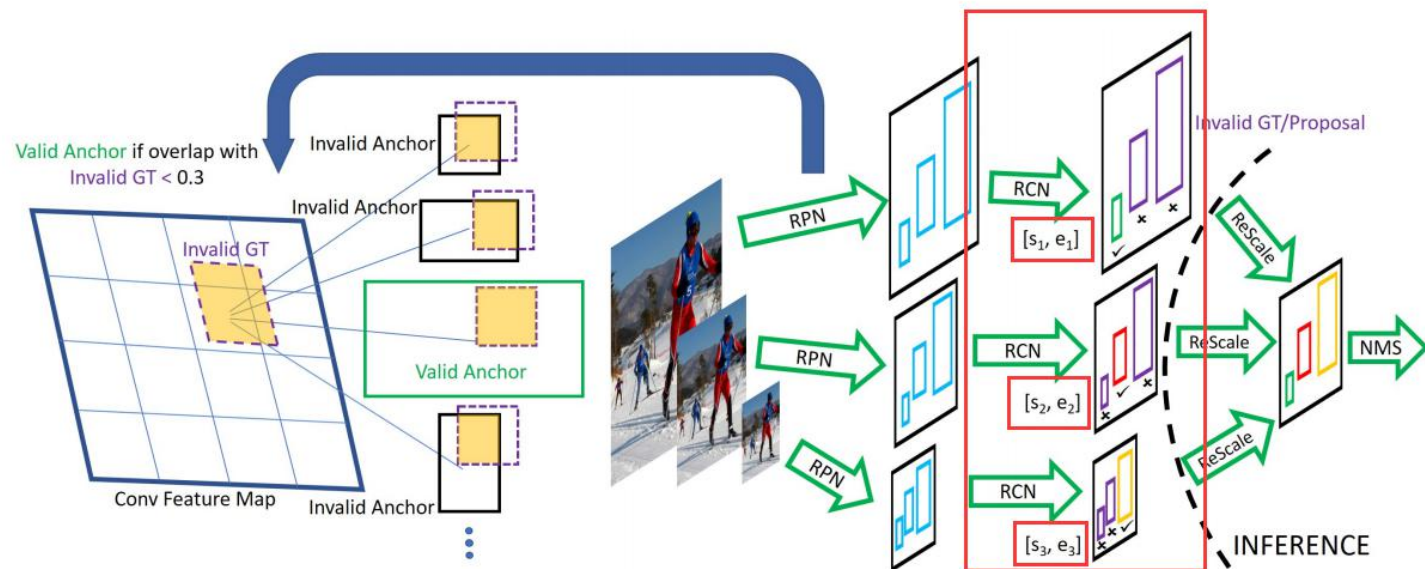
where n is the total number of sample points used in the representation. In our work, n is set to 9 by default.

RepPoints refinement Progressively refining the bounding box localization and feature extraction is important for the success of multi-stage object detection methods. For RepPoints, the refinement can be expressed simply as

$$\mathcal{R}_r = \{(x_k + \Delta x_k, y_k + \Delta y_k)\}_{k=1}^n, \quad (5)$$

Center point based initial object representation. While predefined anchors dominate the representation of objects in the initial stage of object detection, we follow YOLO [31] and DenseBox [17] by using center points as the initial representation of objects, which leads to an anchor-free object detector.

SNIP与TridentNet中的valid the area of an RoI range



这里的range没有给出确切的计算公式，查看源码后发现有直接给出的，不清楚这样的range可信度

snip中的range

$\text{valid_ranges} = [(0, 80), (32, 160), (120,)]$

tridentnet的range

$\text{valid_ranges} = [(0, 90), (30, 160), (90, -1)]$

此range的范围与ATSS的做法有点相似，不过一个是自适应IoU阈值，这里是对三个scale设置valid IoU range，这两种方法的对比效果。

我需要做的

- 1.试着跑一个anchor-free的代码，如RepPoints；去深入理解anchor free方法的一些不懂的细节
- 2.几个比较具有代表性的论文还没有看完，接下来把它们了解清楚
- Trident：三叉戟网络
- DenseBox： RepPoints中说用到DenseBox和yolo的方法来初始化RepPoints，需要结合RepPoints的代码来把RepPoints Generate、RepPoints Learning看明白。