深度学习-实例分割篇

作者：神秘的wz

Mask R-CNN

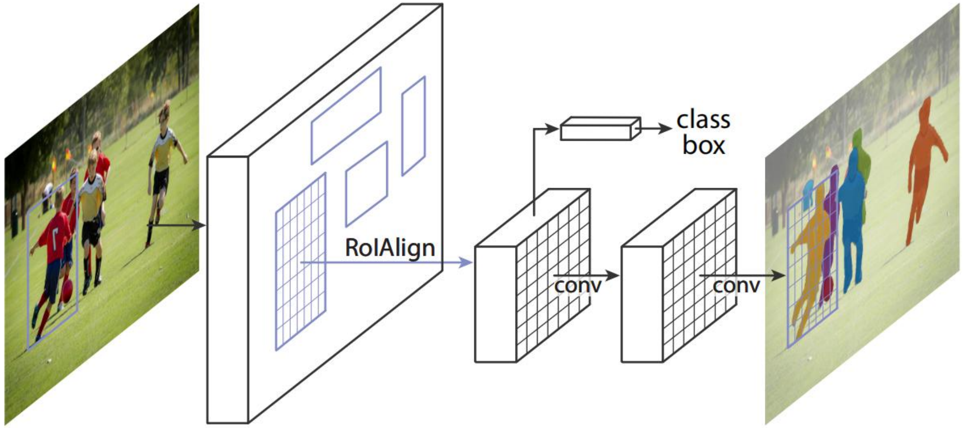
Mask R-CNN

ICCV 2017

Marr Prize

Kaiming He   Georgia Gkioxari   Piotr Dollár   Ross Girshick

Facebook AI Research (FAIR)

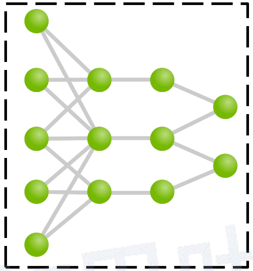


论文下载地址：https://arxiv.org/abs/1703.06870

推荐博文：https://blog.csdn.net/qq_37541097/article/details/123754766

Mask R-CNN

✓ 图像分类



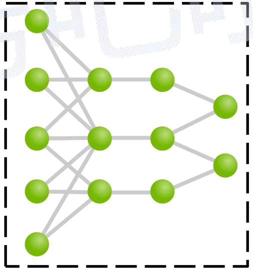
郁金香： 80%

玫瑰花： 10%

向日葵： 5%

鸡蛋花： 5%

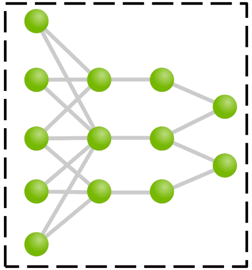
✓ 目标检测



狗: 98%
[50, 20], [80, 100]

猫: 92%
[90, 50], [120, 90]

✓ 语义分割

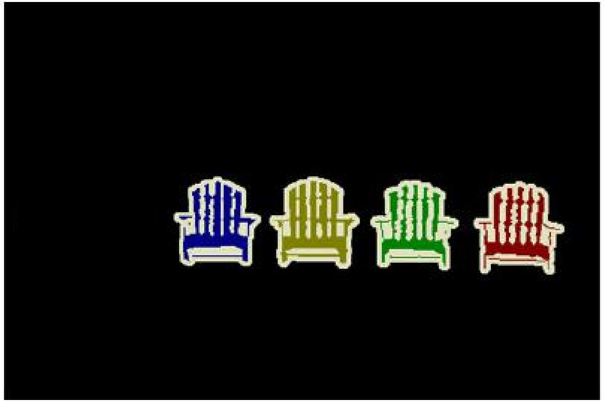


学习本视频之前需要掌握Faster R-CNN、FPN以及FCN的相关知识

Mask R-CNN

原图

实例分割



语义分割

(b) Segmentation

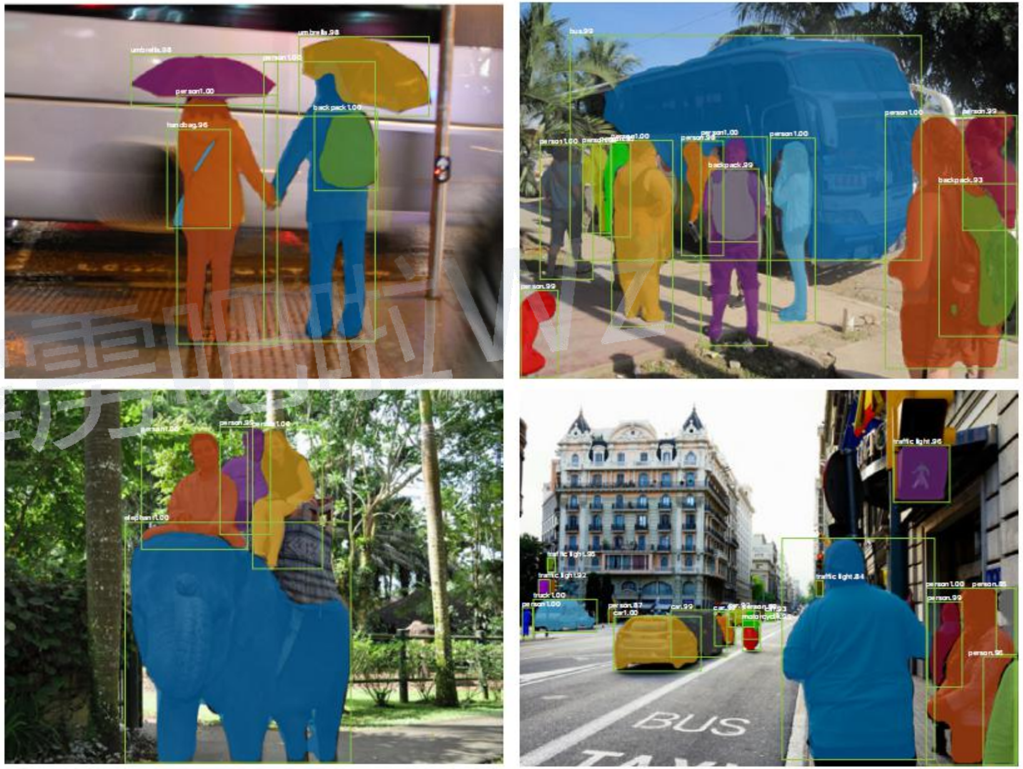
Mask R-CNN

Mask R-CNN不仅能够同时进行目标检测与分割，还能很容易地扩展到其他任务中。

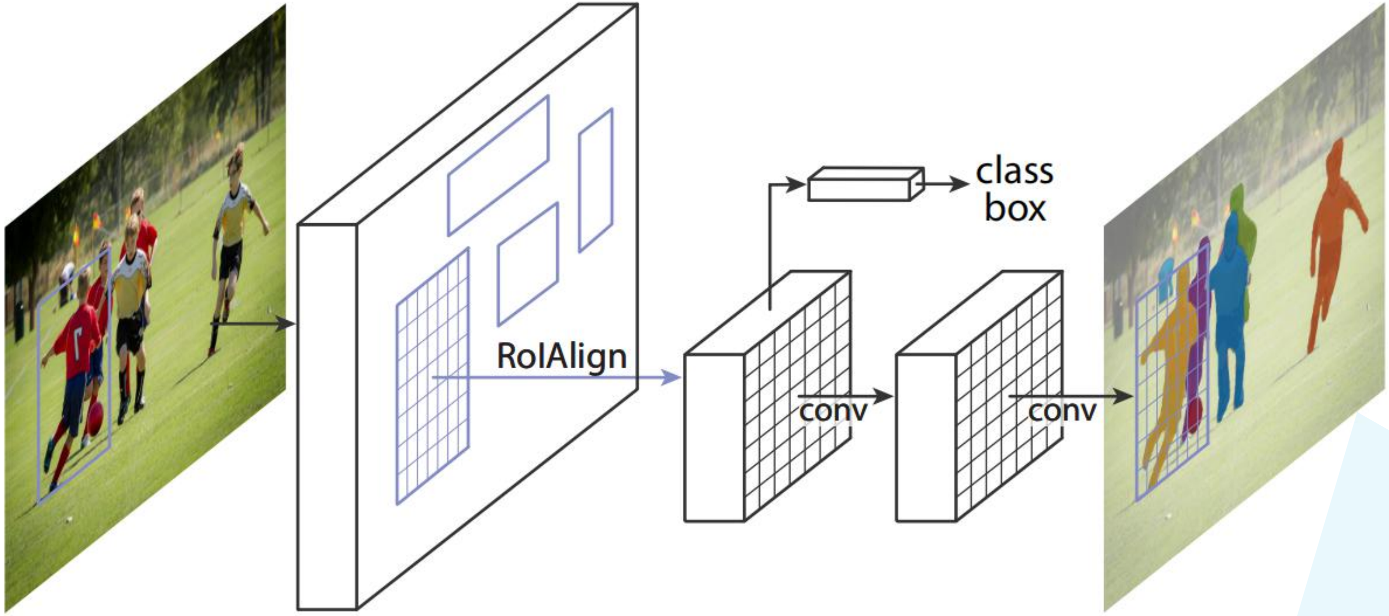
Figure 7. Keypoint detection results on COCO `test` using Mask R-CNN (ResNet-50-FPN), with person segmentation masks predicted from the same model. This model has a keypoint AP of 63.1 and runs at 5 fps.

Mask R-CNN

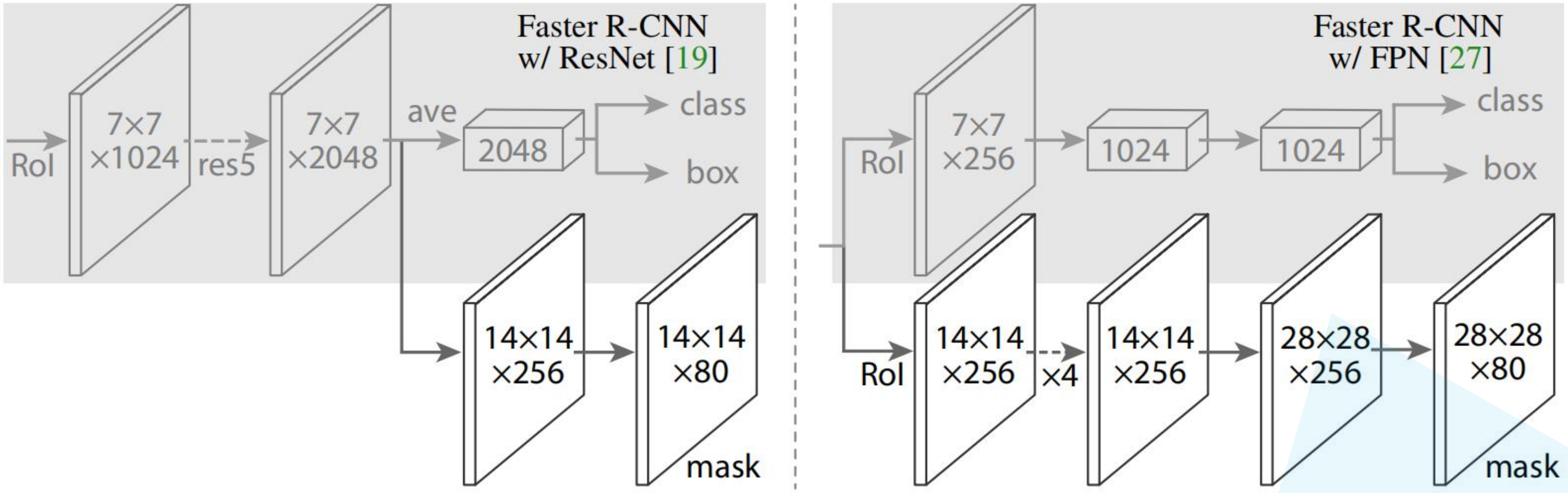
目录



Mask R-CNN



Mask R-CNN



Faster R-CNN
w/ ResNet [19]
RoI
7×7
×1024
res5
7×7
×2048
ave
2048
class
box
14×14
×256
14×14
×80
mask
Faster R-CNN
w/ FPN [27]
RoI
7×7
×256
1024
1024
class
box
RoI
14×14
×256
×4
14×14
×256
28×28
×256
28×28
×80
mask

RoIAlign

	AP	AP_{50}	AP_{75}	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}
RoIPool	23.6	46.5	21.6	28.2	52.7	26.9
RoIAlign	**30.9**	**51.8**	**32.1**	**34.0**	**55.3**	**36.4**
	+7.3	*+ 5.3*	*+10.5*	*+5.8*	*+2.6*	*+9.5*

(d) **RoIAlign** (ResNet-50-**C5**, *stride 32*): Mask-level and box-level AP using *large-stride* features. Misalignments are more severe than with stride-16 features (Table 2c), resulting in big accuracy gaps.

Mask R-CNN

RoIPooling

特征层相对原图步距为32

-1.5256	-0.7502	-0.6540	-1.6095	-0.1002	-0.6092
-0.9798	-1.6091	-0.7121	0.3037	-0.7773	-0.2515
-0.2223	1.6871	0.2284	0.4676	-0.6970	-1.1608
0.6995	0.1991	0.1991	0.0457	0.1530	-0.4757
-1.8821	-0.7765	2.0242	-0.0865	2.3571	-1.0373
1.5748	-0.6298	2.4070	0.2786	0.2468	1.1843

左上角点（原图）：

[10, 10]

$$\left\lfloor \frac{10}{32} \right\rfloor = 0$$

[0, 0]

（特征层）

-1.5256	-0.7502	-0.6540	-1.6095	-0.1002
-0.9798	-1.6091	-0.7121	0.3037	-0.7773
-0.2223	1.6871	0.2284	0.4676	-0.6970
0.6995	0.1991	0.1991	0.0457	0.1530
-1.8821	-0.7765	2.0242	-0.0865	2.3571

右下角点（原图）：

[124, 124]

$$\left\lfloor \frac{124}{32} \right\rfloor = 4$$

[4, 4]

（特征层）

Mask R-CNN

RoIAlign

特征层相对原图步距为32

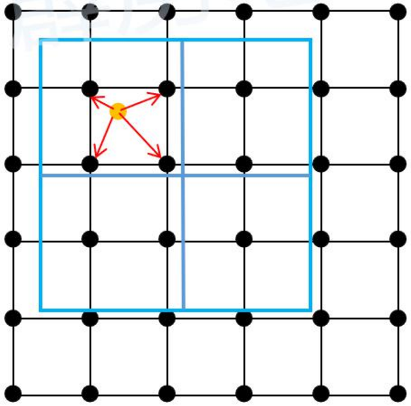
-1.5256	-0.7502	-0.6540	-1.6095	-0.1002	-0.6092
-0.9798	-1.6091	-0.7121	0.3037	-0.7773	-0.2515
-0.2223	1.6871	0.2284	0.4676	-0.6970	-1.1608
0.6995	0.1991	0.1991	0.0457	0.1530	-0.4757
-1.8821	-0.7765	2.0242	-0.0865	2.3571	-1.0373
1.5748	-0.6298	2.4070	0.2786	0.2468	1.1843

左上角点（原图）：

[10, 10]

$$\frac{10}{32} = 0.3125$$

sampling ratio=1



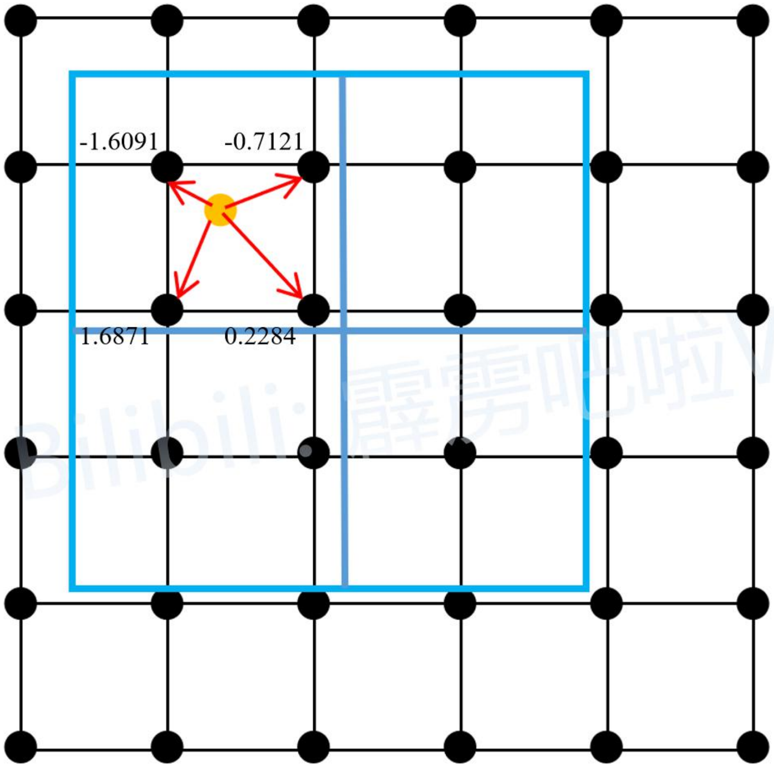
右下角点（原图）：

[124, 124]

$$\frac{124}{32} = 3.875$$

Mask R-CNN

RoIAlign



$$x = 0.3125 + \frac{3.875 - 0.3125}{4} = 1.203125$$

$$y = 0.3125 + \frac{3.875 - 0.3125}{4} = 1.203125$$

$$u = v = 0.20315$$

$$f_1 = -1.6091 \qquad f_2 = -0.7121$$

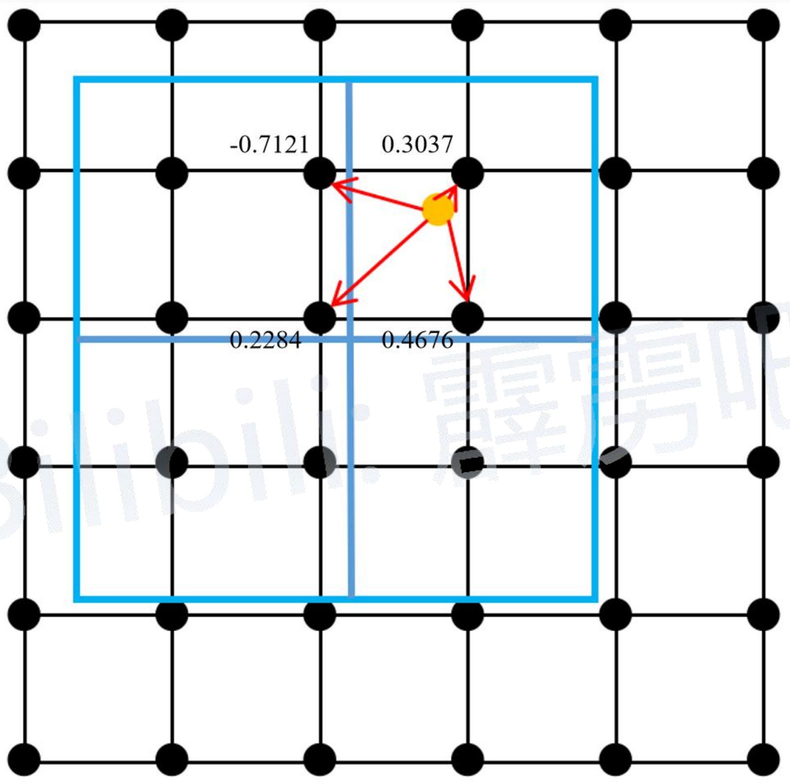
$$f_3 = 1.6871 \qquad f_4 = 0.2284$$

$$f = (1-u)(1-v)f_1 + u(1-v)f_2 + v(1-u)f_3 + uvf_4$$
$$\approx -0.8546$$

双线性插值：https://blog.csdn.net/qq_37541097/article/details/112564822

Mask R-CNN

RoIAlign



$$x = 0.3125 + \frac{3.875 - 0.3125}{4} \times 3 = 2.984375$$

$$y = 0.3125 + \frac{3.875 - 0.3125}{4} = 1.203125$$

$u = 0.984375 \quad v = 0.20315$

$f_1 = -0.7121 \quad f_2 = 0.3037$

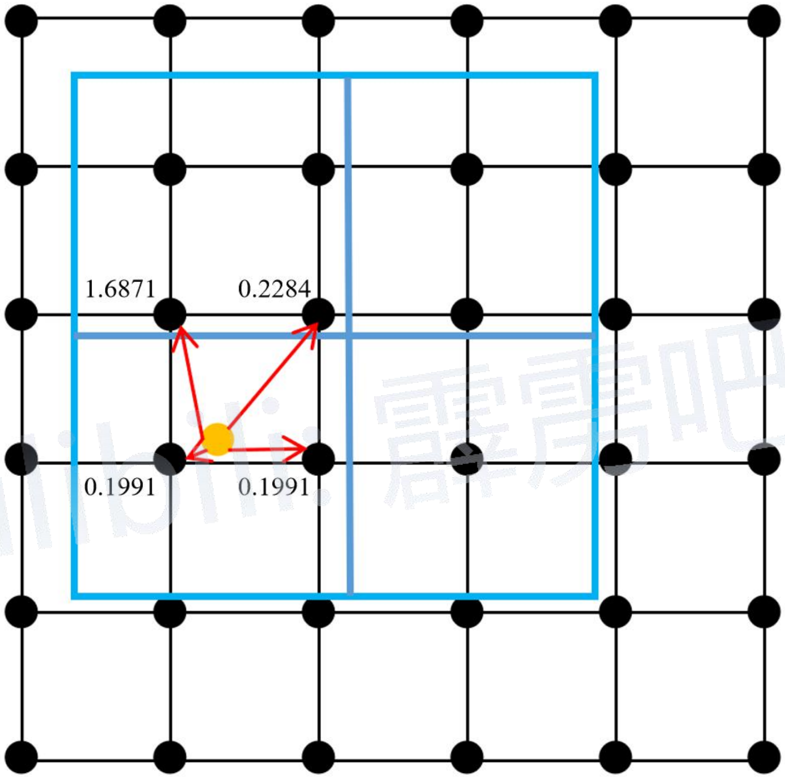
$f_3 = 0.2284 \quad f_4 = 0.4676$

$$f = (1-u)(1-v)f_1 + u(1-v)f_2 + v(1-u)f_3 + uvf_4$$
$$\approx 0.3236$$

双线性插值：https://blog.csdn.net/qq_37541097/article/details/112564822

Mask R-CNN

RoIAlign



$$x = 0.3125 + \frac{3.875 - 0.3125}{4} = 1.203125$$

$$y = 0.3125 + \frac{3.875 - 0.3125}{4} \times 3 = 2.984375$$

$u = 0.20315 \quad v = 0.984375$

$f_1 = 1.6871 \quad f_2 = 0.2284$

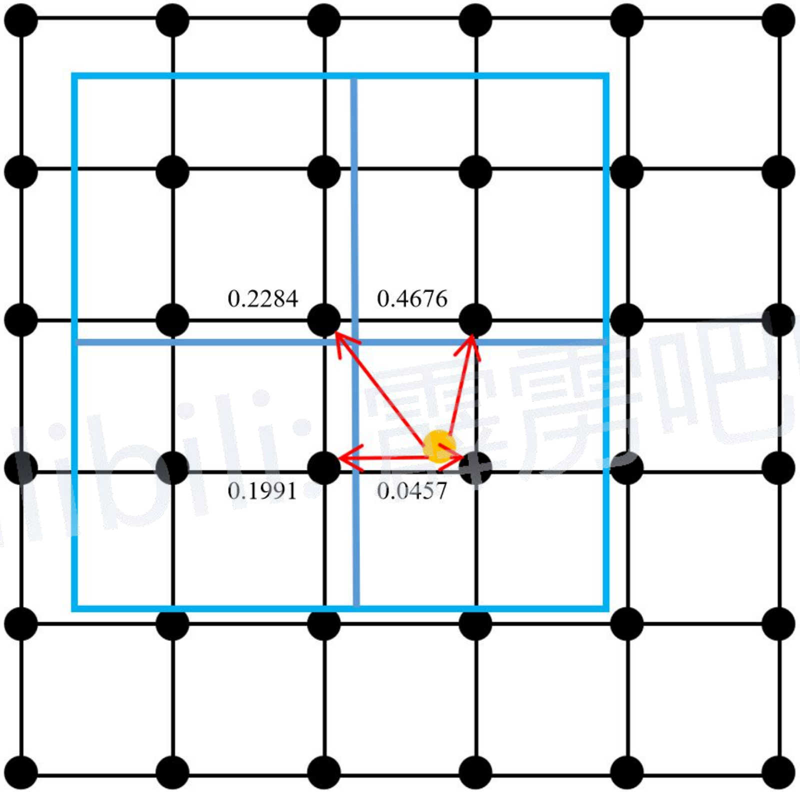
$f_3 = 0.1991 \quad f_4 = 0.1991$

$$f = (1-u)(1-v)f_1 + u(1-v)f_2 + v(1-u)f_3 + uvf_4$$
$$\approx 0.2177$$

双线性插值：https://blog.csdn.net/qq_37541097/article/details/112564822

Mask R-CNN

RoIAlign



$$x = 0.3125 + \frac{3.875 - 0.3125}{4} \times 3 = 2.984375$$

$$y = 0.3125 + \frac{3.875 - 0.3125}{4} \times 3 = 2.984375$$

$u = 0.984375 \quad v = 0.984375$

$f_1 = 0.2284 \quad f_2 = 0.4676$

$f_3 = 0.1991 \quad f_4 = 0.0457$

$$f = (1-u)(1-v)f_1 + u(1-v)f_2 + v(1-u)f_3 + uvf_4 \approx 0.0546$$

双线性插值：https://blog.csdn.net/qq_37541097/article/details/112564822

Mask R-CNN

RoIAlign

We note that the results are not sensitive to the exact sampling locations, or how many points are sampled, as long as no quantization is performed.

最后作者在论文中有提到，关于最终的采样结果对采样点位置，以及采样点的个数并不敏感。

Mask R-CNN

Mask分支

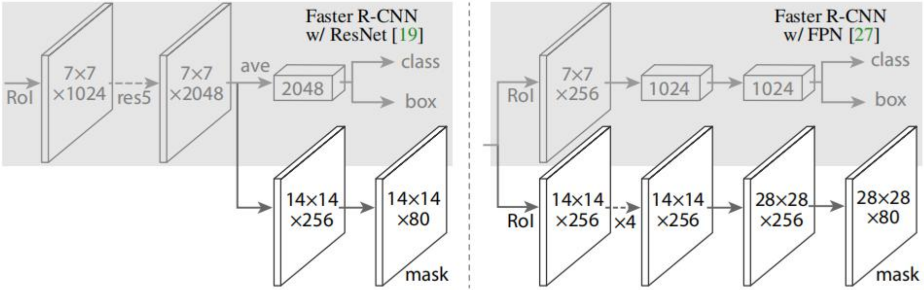
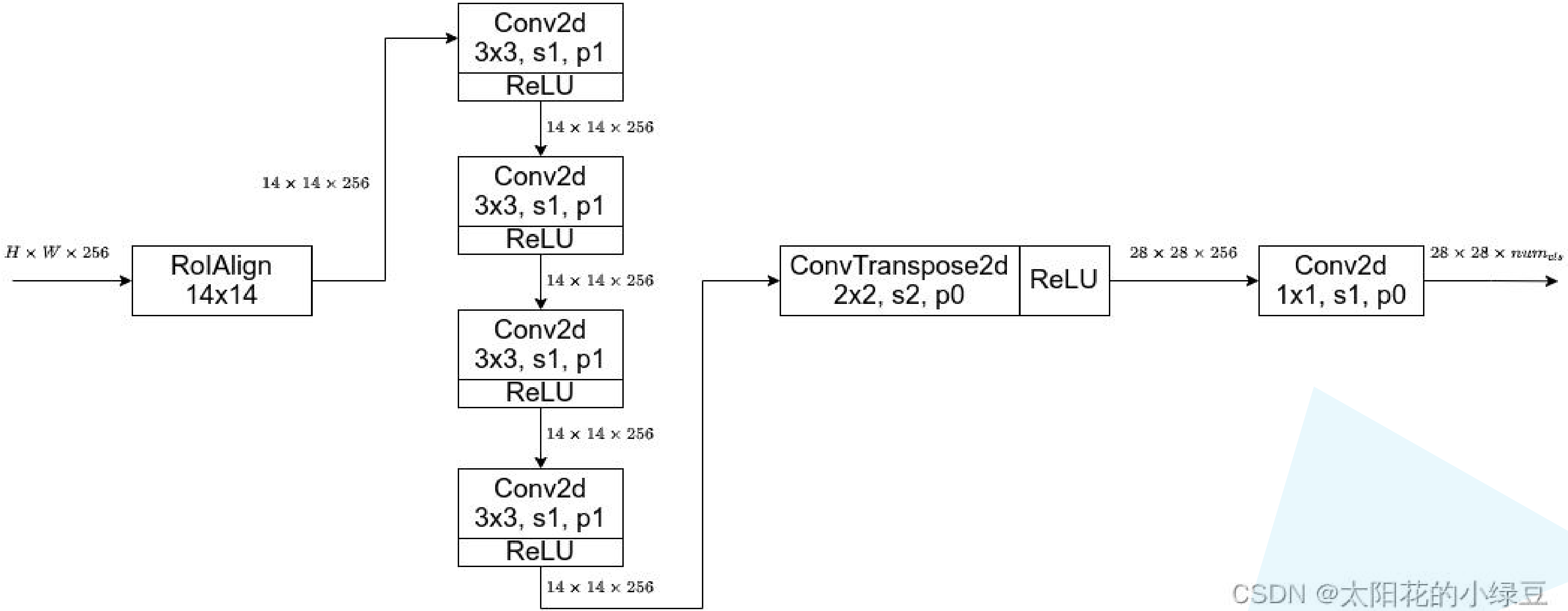


Figure 4. **Head Architecture**: We extend two existing Faster R-CNN heads [19, 27]. Left/Right panels show the heads for the ResNet C4 and FPN backbones, from [19] and [27], respectively, to which a mask branch is added. Numbers denote spatial resolution and channels. Arrows denote either conv, deconv, or *fc* layers as can be inferred from context (conv preserves spatial dimension while deconv increases it). All convs are 3×3, except the output conv which is 1×1, deconvs are 2×2 with stride 2, and we use ReLU [31] in hidden layers. *Left*: 'res5' denotes ResNet's fifth stage, which for simplicity we altered so that the first conv operates on a 7×7 RoI with stride 1 (instead of 14×14 / stride 2 as in [19]). *Right*: '×4' denotes a stack of four consecutive convs.

不共用RoIAlign

Mask R-CNN

Mask分支



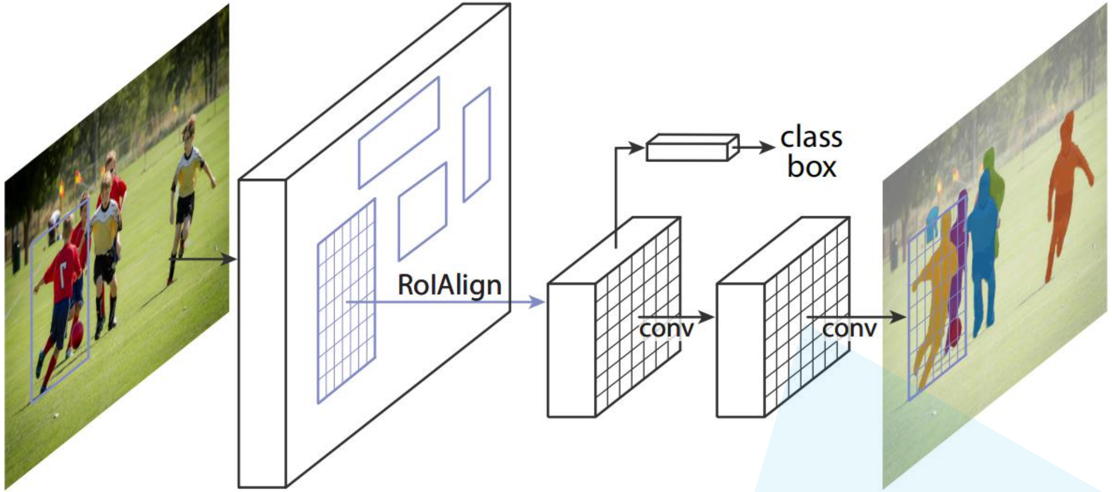
Mask R-CNN

Mask分支

在Mask R-CNN中，对预测Mask以及Class进行解耦

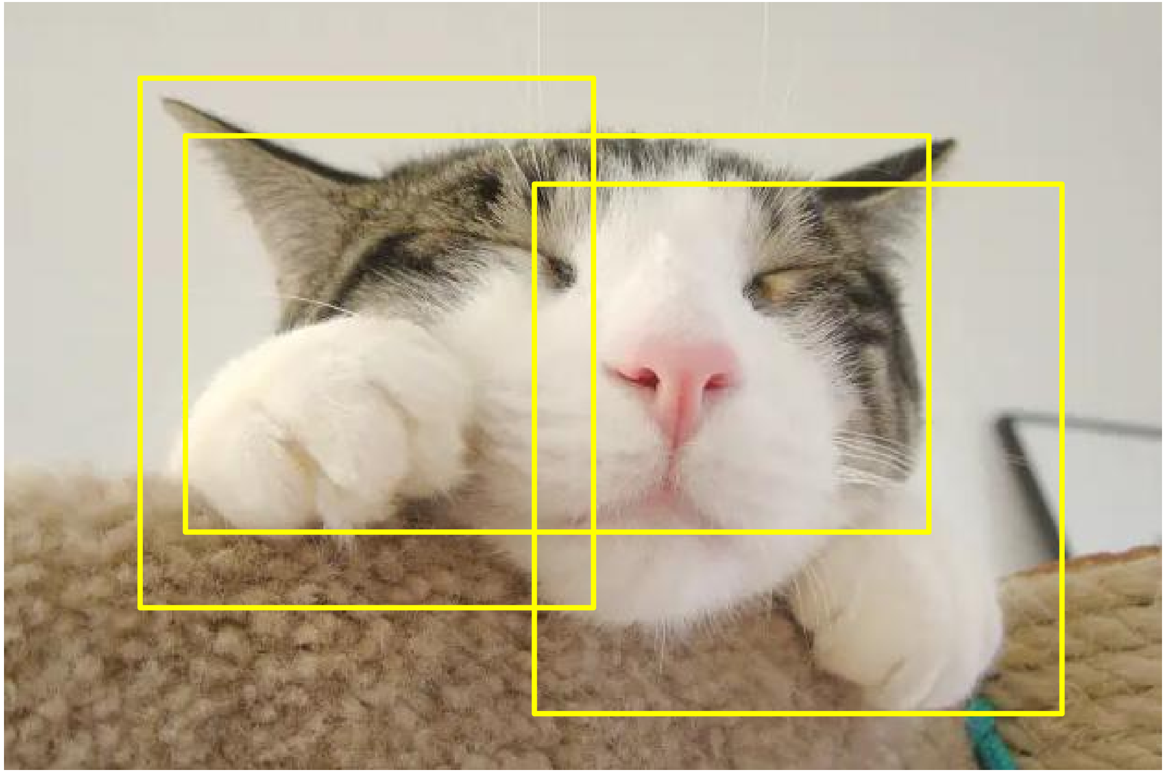
	AP	AP_{50}	AP_{75}
softmax	24.8	44.1	25.1
sigmoid	**30.3**	**51.2**	**31.5**
	+5.5	*+7.1*	*+6.4*

(b) **Multinomial *vs*. Independent Masks** (ResNet-50-C4): *Decoupling* via per-class binary masks (sigmoid) gives large gains over multinomial masks (softmax).



训练网络的时候输入Mask分支的目标是由RPN提供的，即Proposals，(正样本)
但在预测的时候输入Mask分支的目标是由Fast R-CNN提供

Mask分支



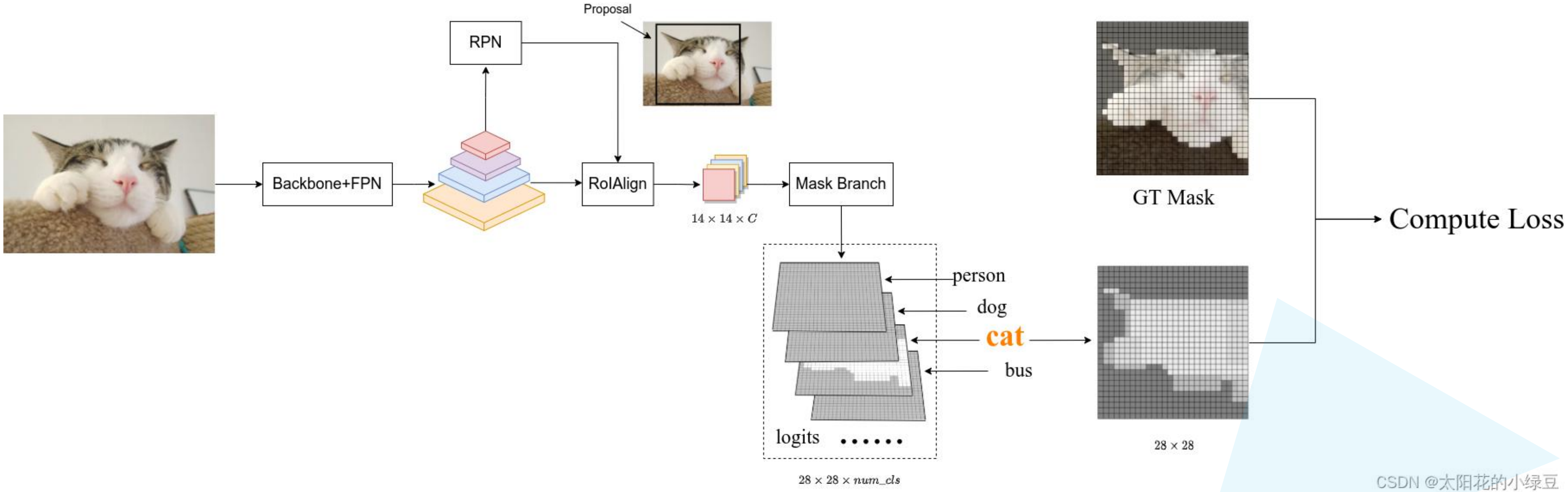
Mask R-CNN损失

$$Loss = L_{rpn} + L_{fast\_rcnn} + L_{mask}$$

Mask R-CNN

Mask分支损失

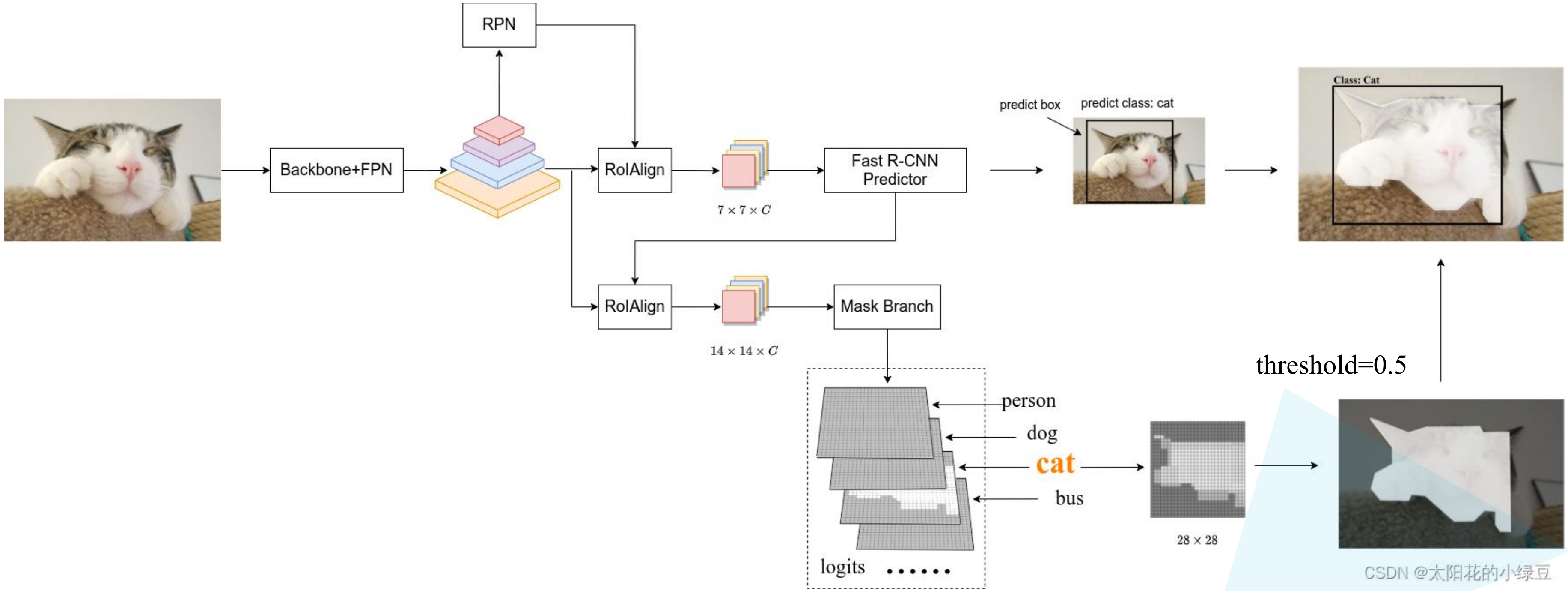
BCELoss (BinaryCrossEntropyLoss)



训练网络的时候输入Mask分支的目标是由RPN提供的，即Proposals，(正样本)
但在预测的时候输入Mask分支的目标是由Fast R-CNN提供

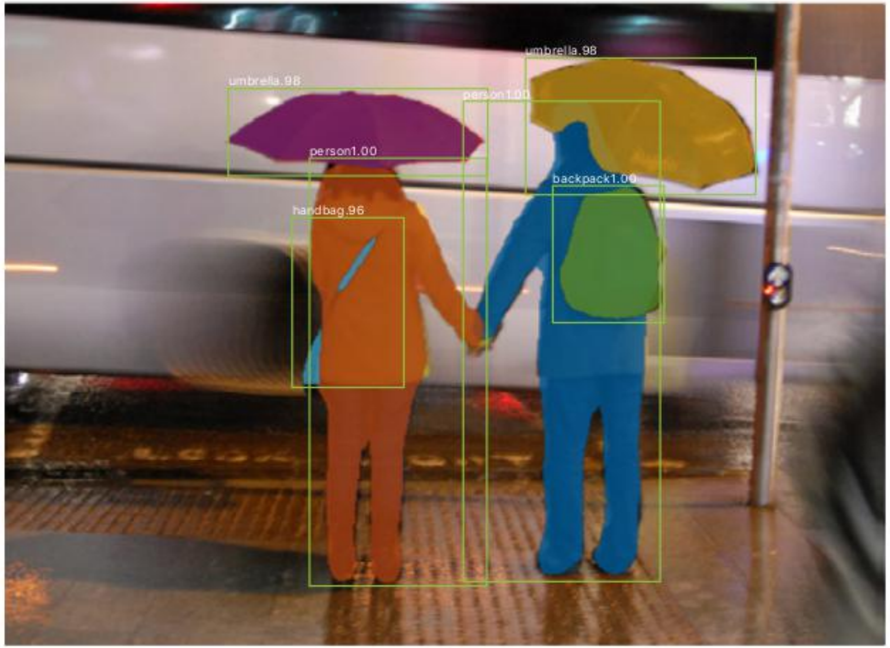
Mask R-CNN

Mask分支预测使用



训练网络的时候输入Mask分支的目标是由RPN提供的，即Proposals，
但在预测的时候输入Mask分支的目标是由Fast R-CNN提供

Mask R-CNN


umbrella.98
umbrella.98
person1.00
person1.00
backpack1.00
handbag.96


bus.99
person1.00
person.99
person1.00
person1.00
backpack.99
person1.00
backpack.93
person.99
CSDN @太阳花的小绿豆

沟通方式

1.github

https://github.com/WZMIAOMIAO/deep-learning-for-image-processing

2.bilibili

https://space.bilibili.com/18161609/channel/index

3.CSDN

https://blog.csdn.net/qq_37541097/article/details/103482003