目标检测性能指标

检测精度

- Precision, Recall, F1 score
- IoU (Intersection over Union)
- P-R curve （Precison-Recall curve）
- AP (Average Precision)
- mAP (mean Average Precision)

检测速度

- 前传耗时
- 每秒帧数 FPS (Frames Per Second)
- 浮点运算量(FLOPS)

混淆矩阵（confusion matrix）

		预测 Prediction	
		Positive	Negative
实际 Actual	True	TP	FN
	False	FP	TN

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

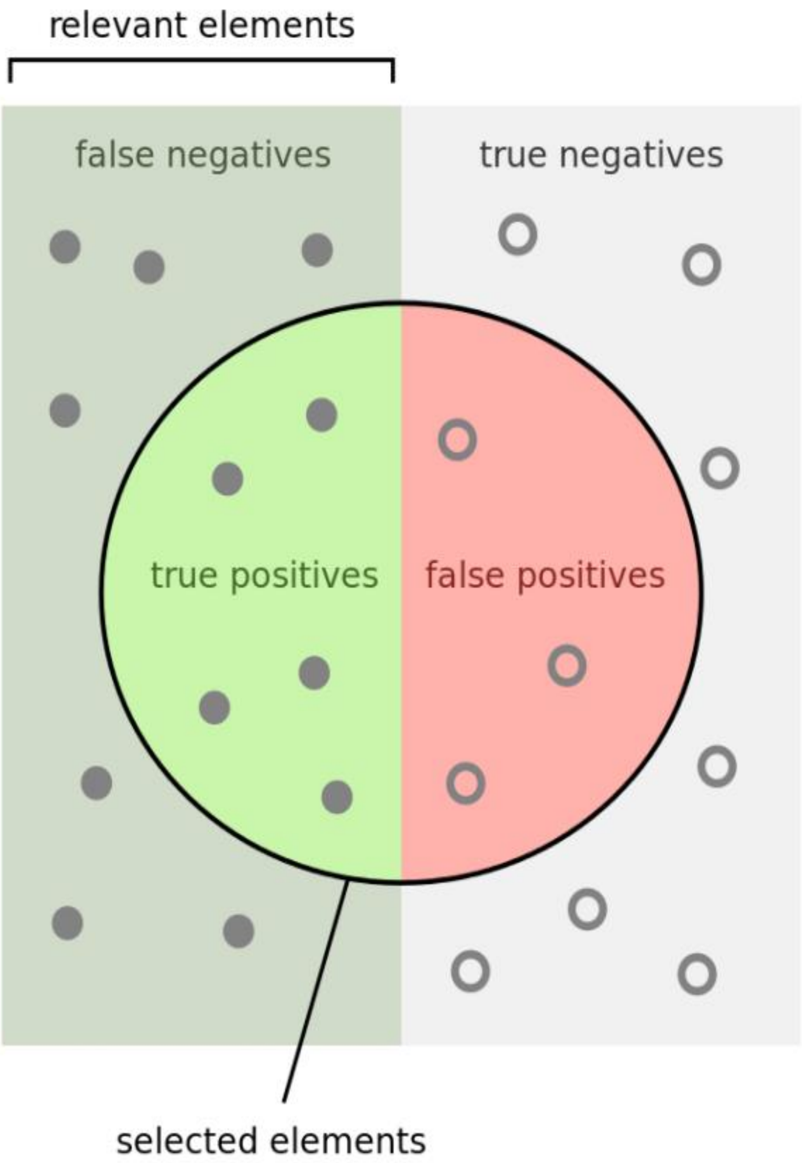
$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

$$F1\ score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

第一位T/F: 表示预测的对错

第二位P/N: 表示预测的结果

- 精度Precision(查准率)是评估预测的准不准（看预测列）
- 召回率Recall（查全率）是评估找的全不全(看实际行)

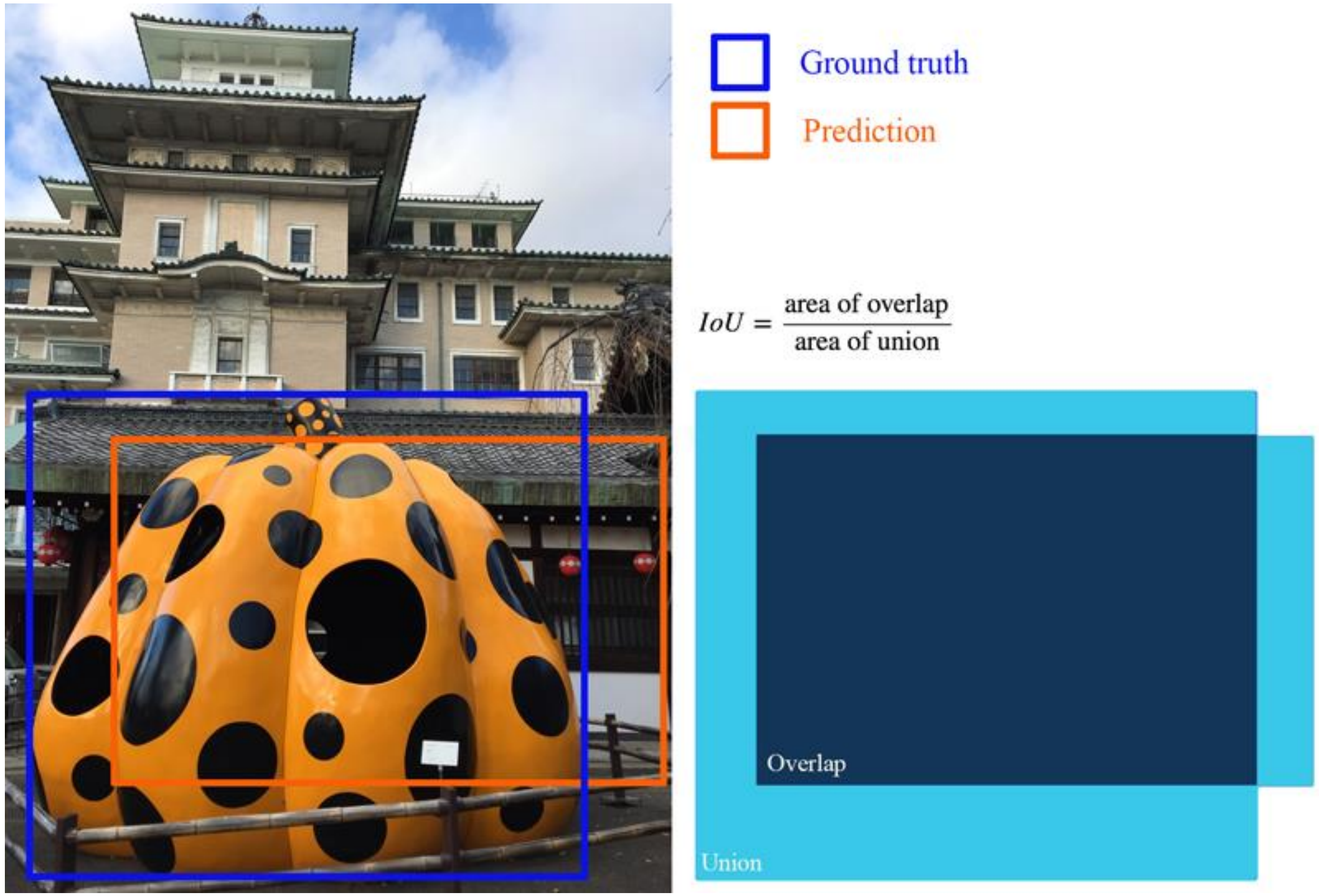


How many selected items are relevant?

How many relevant items are selected?

IoU (Intersection over Union)

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$



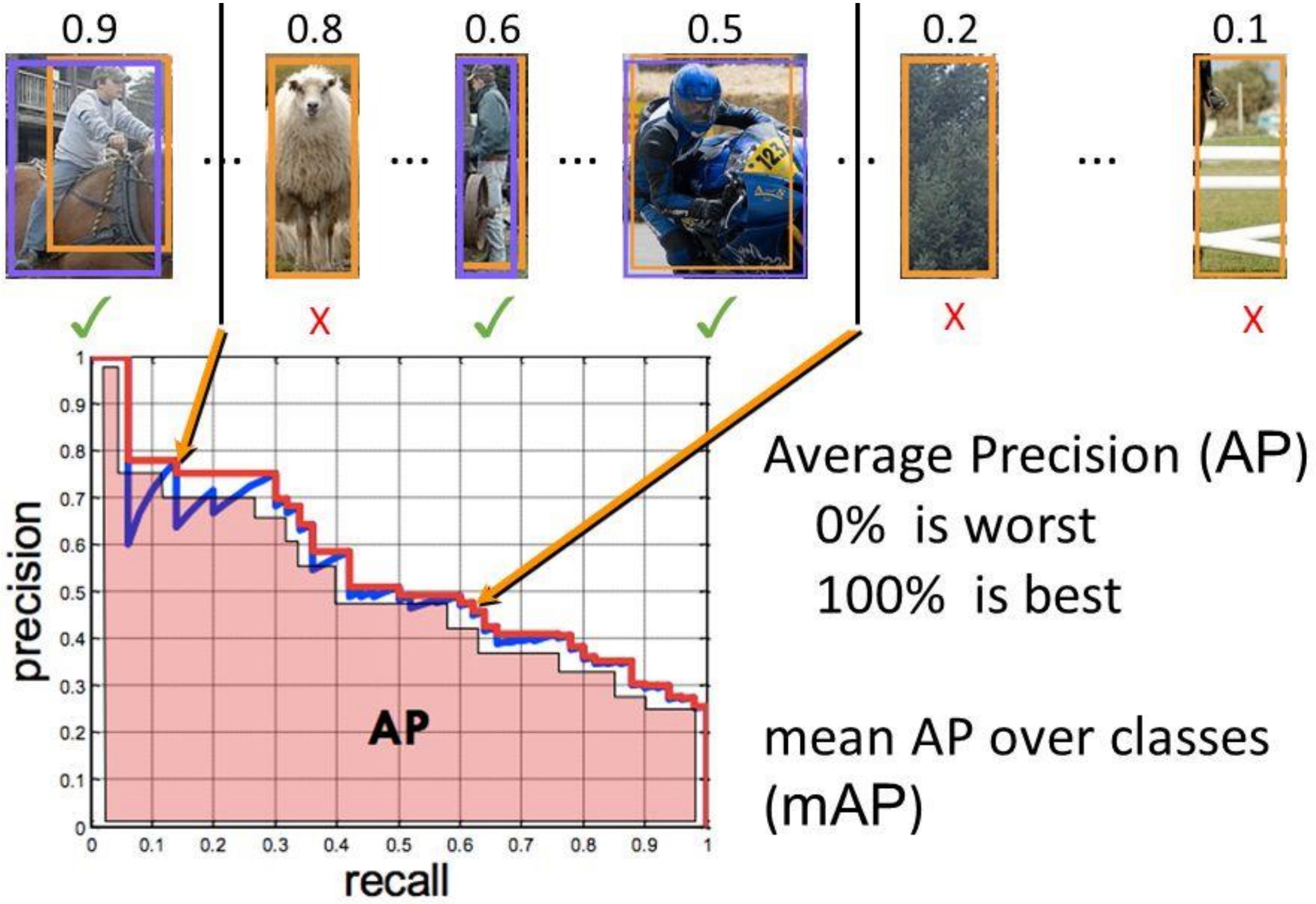
An IoU of 1 implies that predicted and the ground-truth bounding boxes perfectly overlap.

You can set a threshold value for the IoU to determine if the object detection is valid or not.

Let’s say you set IoU to 0.5, in that case

- **if IoU ≥0.5,** classify the object detection as **True Positive(TP)**
- **if IoU <0.5**, then it is a wrong detection and classify it as **False Positive(FP)**
- **When a ground truth is present in the image and model failed to detect the object,** classify it as **False Negative(FN).**
- **True Negative (TN**): TN is every part of the image where we did not predict an object. This metrics is not useful for object detection, hence we ignore TN.

- AP衡量的是学习出来的模型在每个类别上的好坏
- mAP衡量的是学出的模型在所有类别上的好坏。mAP就是取所有类别上AP的平均值。



net	data	mAP	areo	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
VGG-16	07+12	73.2	76.5	79.0	70.9	65.5	52.1	83.1	84.7	86.4	52.0	81.9	65.7	84.8	84.6	77.5	76.7	38.8	73.6	73.9	83.0	72.6
ResNet-101	07+12	76.4	79.8	80.7	76.2	68.3	55.9	85.1	85.3	**89.8**	56.7	87.8	69.4	88.3	88.9	80.9	78.4	41.7	78.6	79.8	85.3	72.0
ResNet-101	COCO+07+12	**85.6**	**90.0**	**89.6**	**87.8**	**80.8**	**76.1**	**89.9**	**89.9**	89.6	**75.5**	**90.0**	**80.7**	**89.6**	**90.3**	**89.1**	**88.7**	**65.4**	**88.1**	**85.6**	**89.0**	**86.8**

AP (Average Precision) in PASCAL VOC challenge

- 对于PASCAL VOC挑战，如果IoU> 0.5，则预测为正样本(TP)。 但是，如果检测到同一目标的多个检测，则视第一个检测为正样本(TP)，而视其余检测为负样本(FP)。

COCO AP & mAP

AP@.50 means the AP with IoU=0.50.

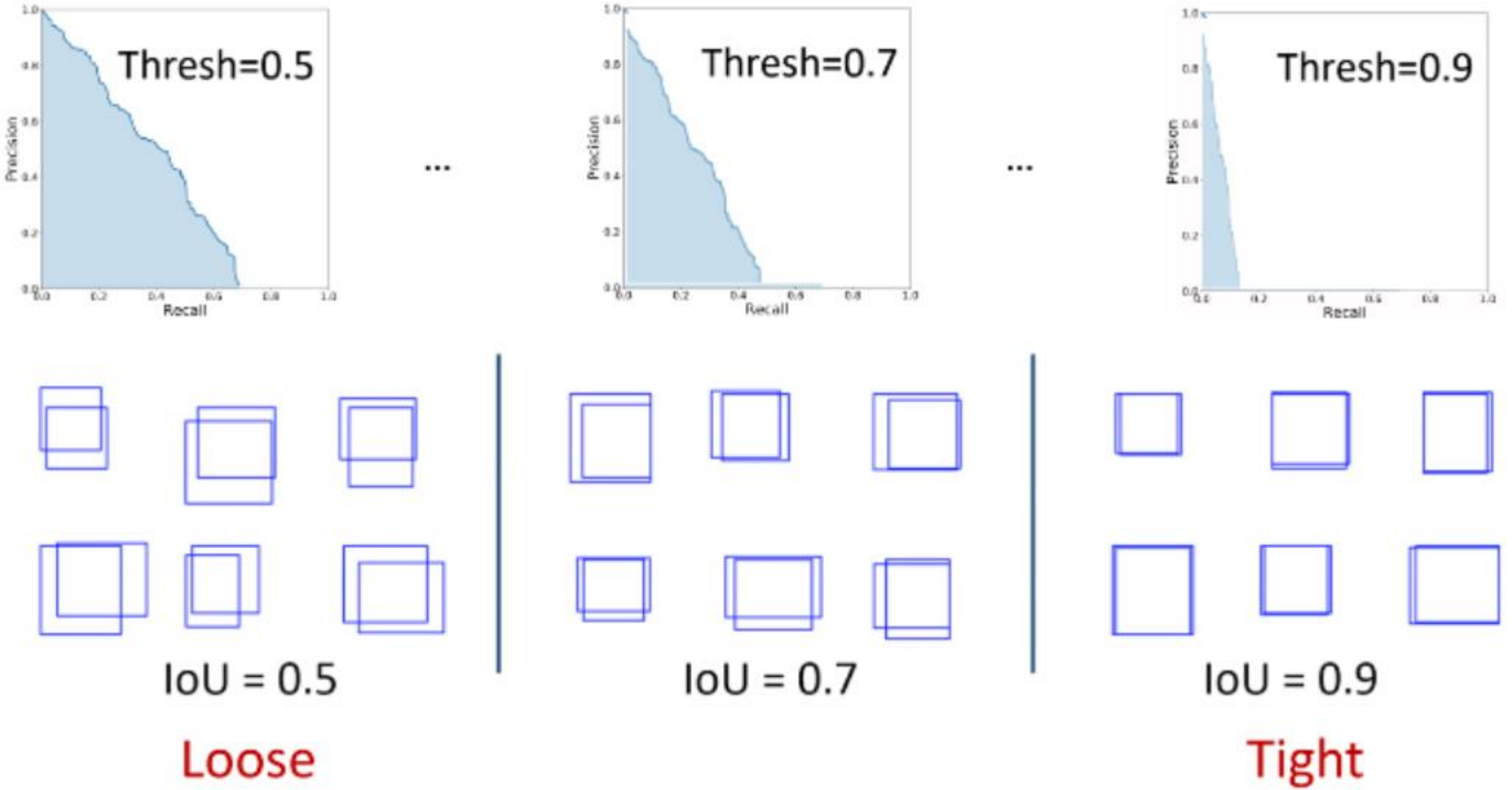
AP@.75 means the AP with IoU=0.75.

For COCO, AP is the average over multiple IoU (the minimum IoU to consider a positive match). **AP@[.5:.95]** corresponds to the average AP for IoU from 0.5 to 0.95 with a step size of 0.05.

For the COCO competition, AP is the average over 10 IoU levels on 80 categories (AP@[.50:.05:.95]: start from 0.5 to 0.95 with a step size of 0.05.

$$mAP_{\text{COCO}} = \frac{mAP_{0.50} + mAP_{0.55} + \ldots + mAP_{0.95}}{10}$$

$$\mathrm{AP}[\mathrm{class}] = \frac{1}{\#\mathrm{thresolds}} \sum_{\mathrm{iou} \in \mathrm{thresholds}} AP[class, iou]$$



COCO AP

Average Precision (AP):

AP	% AP at IoU=.50:.05:.95 **(primary challenge metric)**
$AP^{IoU=.50}$	% AP at IoU=.50 (PASCAL VOC metric)
$AP^{IoU=.75}$	% AP at IoU=.75 (strict metric)

AP Across Scales:

AP^{small}	% AP for small objects: area < 32^2
AP^{medium}	% AP for medium objects: 32^2 < area < 96^2
AP^{large}	% AP for large objects: area > 96^2

Average Recall (AR):

$AR^{max=1}$	% AR given 1 detection per image
$AR^{max=10}$	% AR given 10 detections per image
$AR^{max=100}$	% AR given 100 detections per image

AR Across Scales:

AR^{small}	% AR for small objects: area < 32^2
AR^{medium}	% AR for medium objects: 32^2 < area < 96^2
AR^{large}	% AR for large objects: area > 96^2

Table 10: Comparison of the speed and accuracy of different object detectors on the MS COCO dataset (test-dev 2017). (Real-time detectors with FPS 30 or higher are highlighted here. We compare the results with batch=1 without using tensorRT.)

Method	Backbone	Size	FPS	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
YOLOv4: Optimal Speed and Accuracy of Object Detection									
YOLOv4	CSPDarknet-53	416	96 (V)	41.2%	62.8%	44.3%	20.4%	44.4%	56.0%
YOLOv4	CSPDarknet-53	512	83 (V)	43.0%	64.9%	46.5%	24.3%	46.1%	55.2%
YOLOv4	CSPDarknet-53	608	62 (V)	**43.5%**	**65.7%**	47.3%	**26.7%**	46.7%	53.3%
EfficientDet: Scalable and Efficient Object Detection [77]									
EfficientDet-D0	Efficient-B0	512	62.5 (V)	33.8%	52.2%	35.8%	12.0%	38.3%	51.2%
EfficientDet-D1	Efficient-B1	640	50.0 (V)	39.6%	58.6%	42.3%	17.9%	44.3%	56.0%
EfficientDet-D2	Efficient-B2	768	41.7 (V)	43.0%	62.3%	46.2%	22.5%	**47.0%**	**58.4%**
EfficientDet-D3	Efficient-B3	896	23.8 (V)	45.8%	65.0%	49.3%	26.6%	49.4%	59.8%
Learning Spatial Fusion for Single-Shot Object Detection [48]									
YOLOv3 + ASFF*	Darknet-53	320	60 (V)	38.1%	57.4%	42.1%	16.1%	41.6%	53.6%
YOLOv3 + ASFF*	Darknet-53	416	54 (V)	40.6%	60.6%	45.1%	20.3%	44.2%	54.1%
YOLOv3 + ASFF*	Darknet-53	608×	45.5 (V)	42.4%	63.0%	**47.4%**	25.5%	45.7%	52.3%
YOLOv3 + ASFF*	Darknet-53	800×	29.4 (V)	43.9%	64.1%	49.2%	27.0%	46.6%	53.4%

AP (Average Precison)

用一个简单的例子来演示平均精度（AP）的计算。 假设数据集中总共有5个苹果。 我们收集模型为苹果作的所有预测，并根据预测的置信水平（从最高到最低）对其进行排名。 第二列表示预测是否正确。 如果它与ground truth匹配并且IoU≥0.5，则是正确的。

Rank	Correct?	Precision	Recall
1	True	1.0	0.2
2	True	1.0	0.4
3	False	0.67	0.4
4	False	0.5	0.4
5	False	0.4	0.4
6	True	0.5	0.6
7	True	0.57	0.8
8	False	0.5	0.8
9	False	0.44	0.8
10	True	0.5	1.0

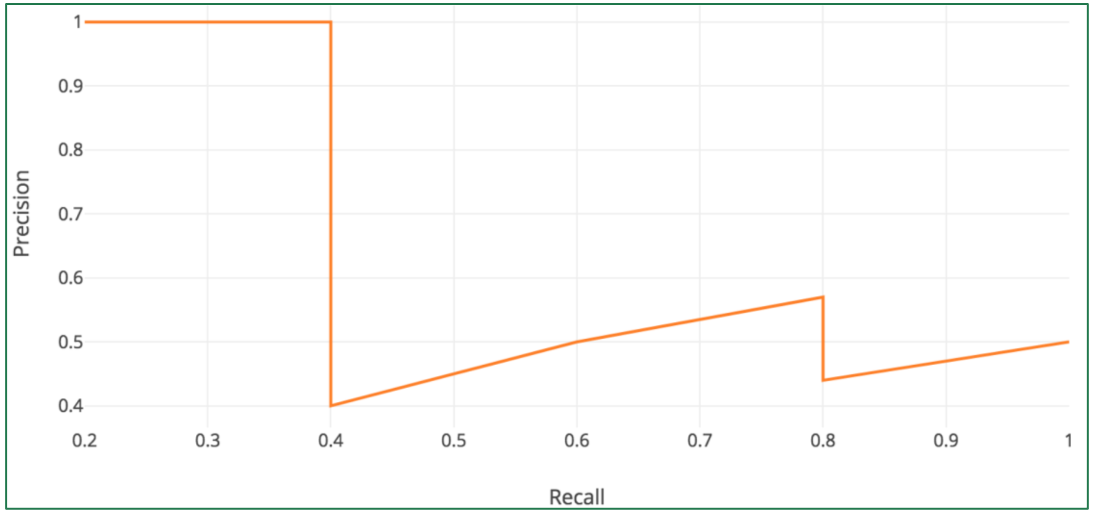
Let’s compute the precision and recall value for the row with rank #3.

Precision is the proportion of TP out of (TP+FP)= 2/3 = 0.67.

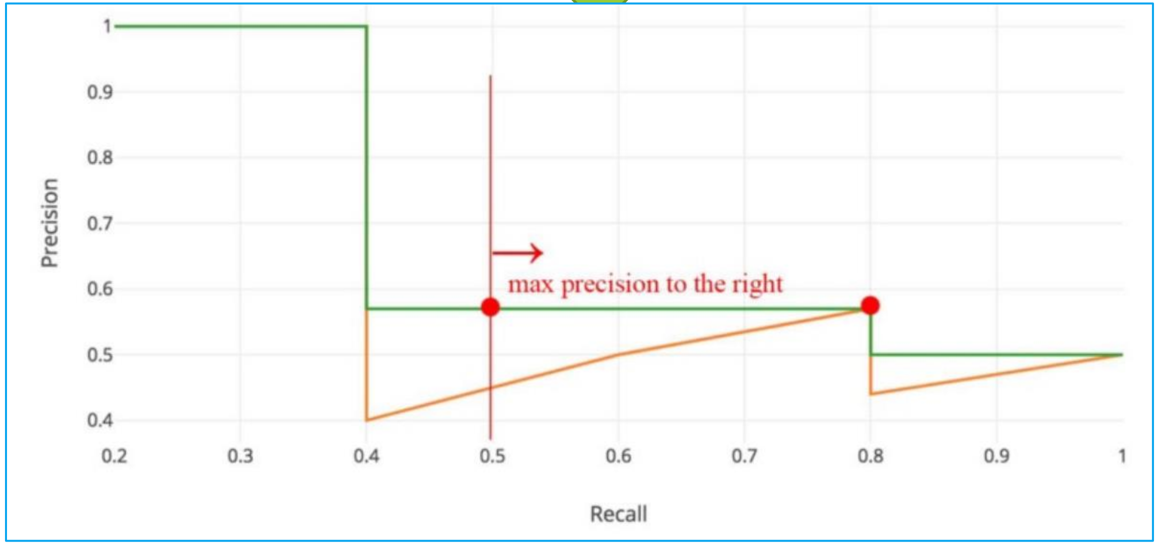
Recall is the proportion of TP out of the possible positives = 2/5 = 0.4.

Recall随着包含更多预测而增加，但Precision会上下波动。

AP (Average Precison)计算



AP在概念上可以视为在precision-recall curve（橙色线）下方的区域。

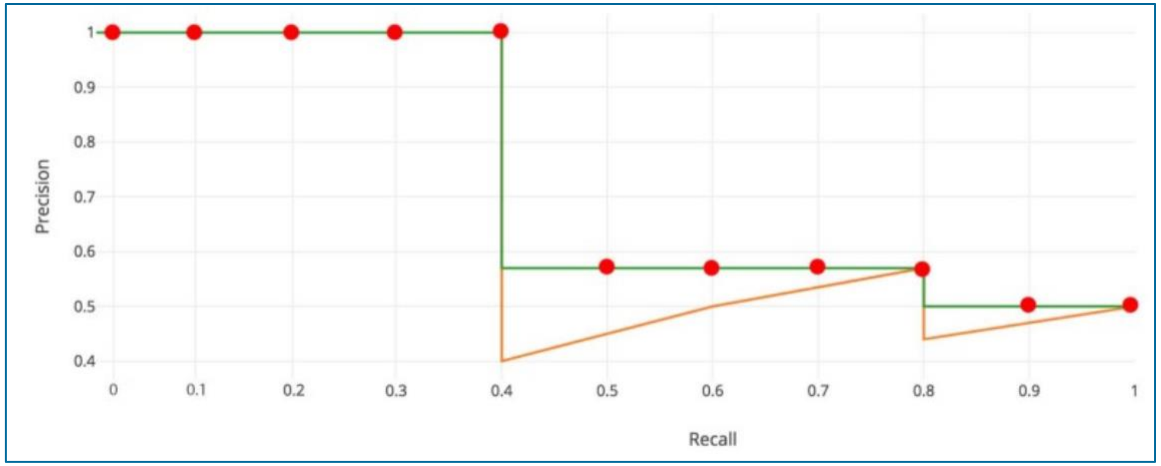


首先通过平滑锯齿形图形来近似这样的计算

We plot the graph with recall value at 0, 0.1, 0.2, ..., 0.9 and 1.0 and we replace the precision value with the maximum precision for any recall ≥ $\tilde{r}$

$$p_{interp}(r) = \max_{\tilde{r} \geq r} p(\tilde{r})$$

AP计算之11点法



AP（平均精度）计算为这11个recall级别的最大精度的平均值： 11-point interpolated average precision

$$AP = \frac{1}{11} \times \left(AP_r(0) + AP_r(0.1) + \ldots + AP_r(1.0)\right)$$

这近似于找到绿色曲线下的总面积并将其除以11。下面是更精确的定义：

$$\begin{aligned} AP &= \frac{1}{11} \sum_{r \in \{0.0,\ldots,1.0\}} AP_r \\ &= \frac{1}{11} \sum_{r \in \{0.0,\ldots,1.0\}} p_{interp}(r) \end{aligned}$$

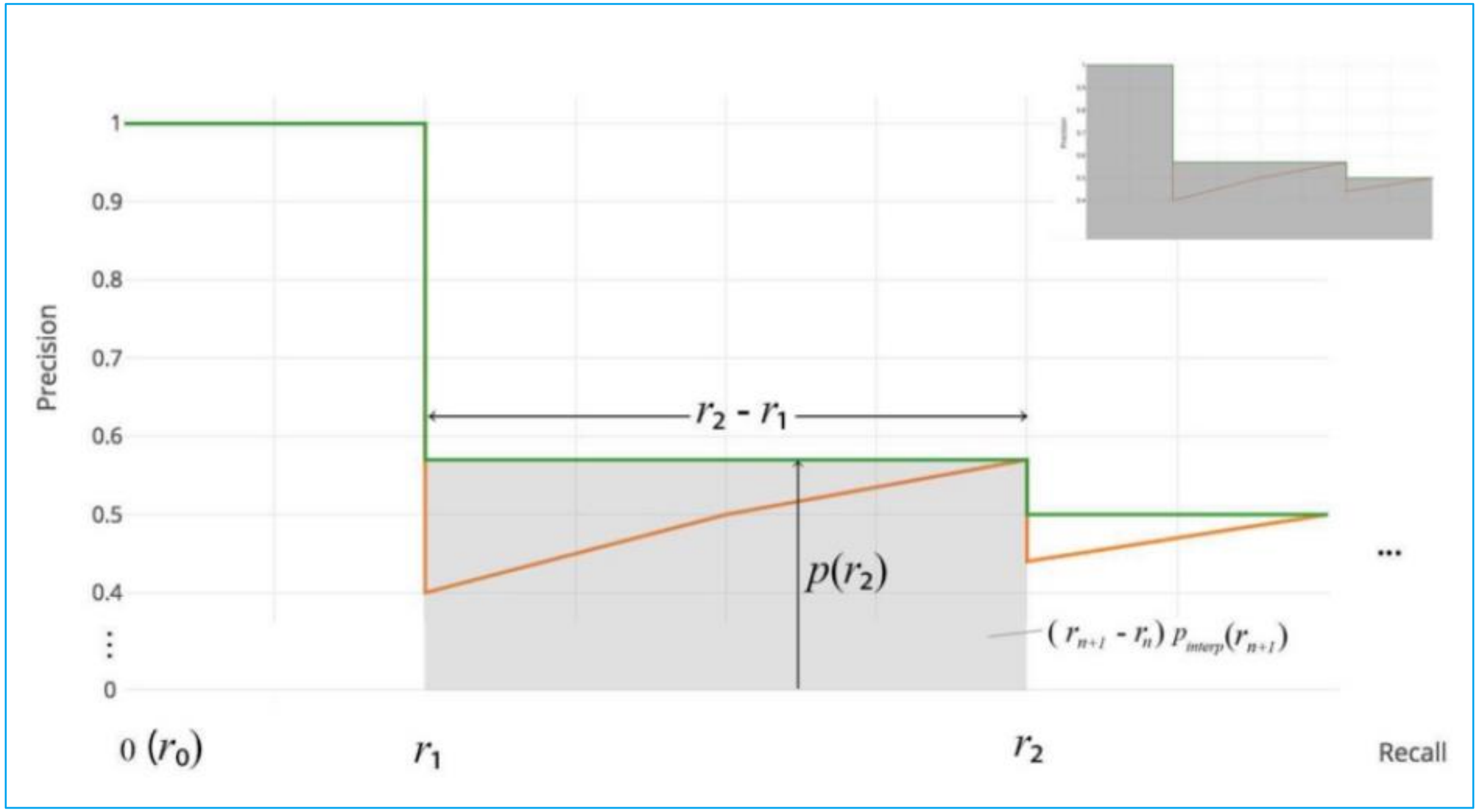
where

$$p_{interp}(r) = \max_{\tilde{r} \geq r} p(\tilde{r})$$

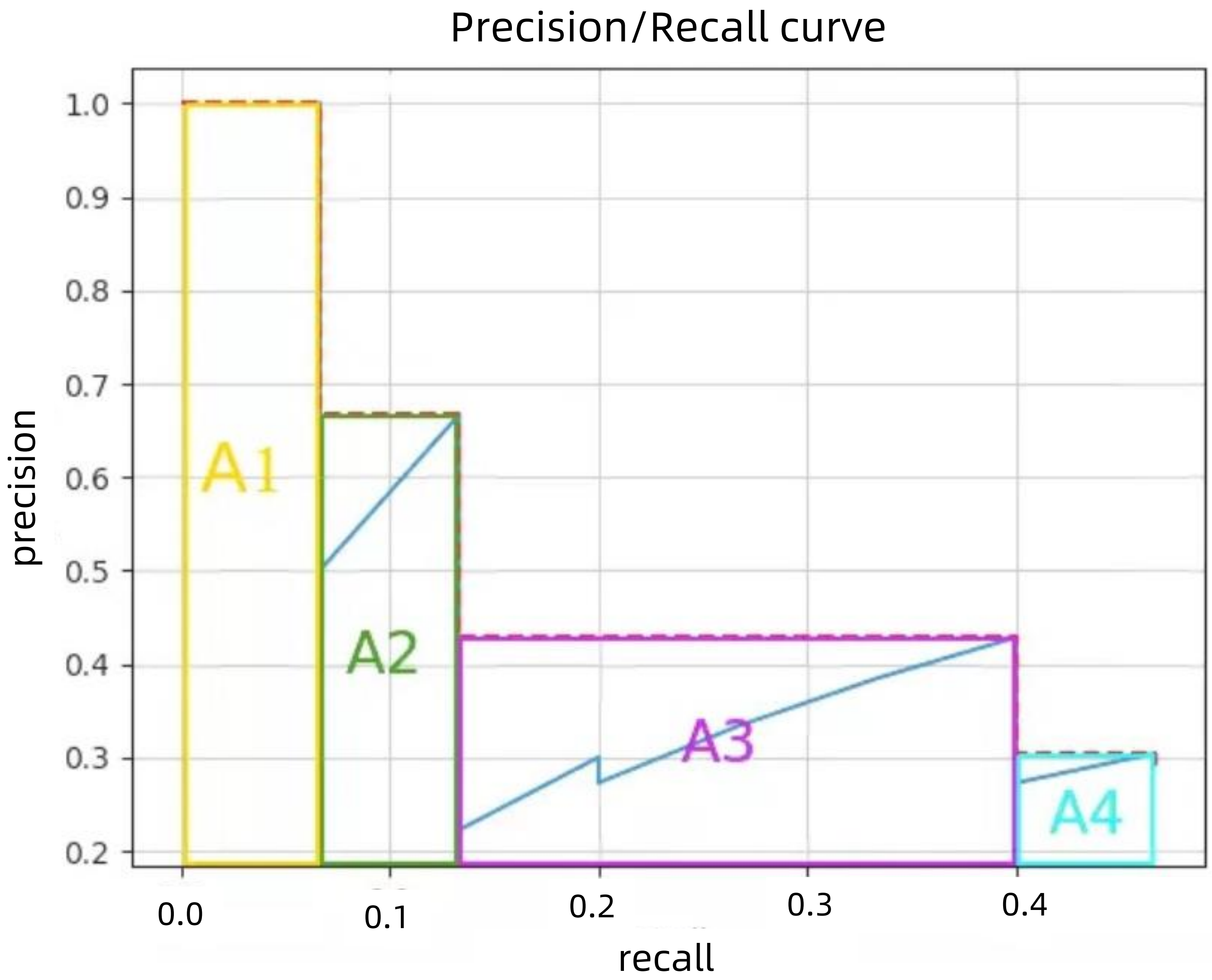
在本例中 AP = $\frac{5 \times 1.0 + 4 \times 0.57 + 2 \times 0.5}{11}$ ≈75.3%

AP计算之积分法 (Area under curve AUC)

PASCAL VOC CHALLENGE自2010年后换了一种计算方法。新的计算方法假设这N个样本中有M个正例，那么我们会得到M个recall值（1/M, 2/M, ..., M/M）,对于每个recall值r，我们可以计算出对应（r' > r）的最大precision，然后对这M个precision值取平均即得到最后的AP值。



AP计算之积分法 (Area under curve AUC)



$$AP = A1 + A2 + A3 + A4$$

小 结

- Pascal VOC2007 uses 11 Recall points on PR curve.
- Pascal VOC2010–2012 uses (all points) Area Under Curve (AUC) on PR curve.
- MS COCO uses 101 Recall points on PR curve as well as different IoU thresholds.

检测速度

- **前传耗时(ms):** 从输入一张图像到输出最终结果所消耗的时间，包括前处理耗时(如图像归一化)、网络前传耗时、后处理耗时(如非极大值抑制)
- **每秒帧数** FPS (Frames Per Second)：每秒钟能处理的图像数量
- **浮点运算量**(FLOPS)：处理一张图像所需要的浮点运算数量, 跟具体软硬件没有关系, 可以公平地比较不同算法之间的检测速度。