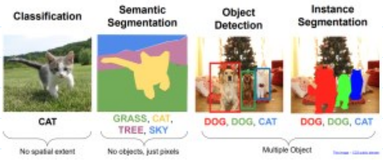
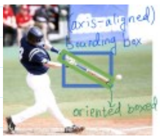
5 Object detection

2024年9月26日 16:13

1 Task:



2. 

Bounding box: visible part

Amodal detection: entire (occluded parts)

3 Intersection over union (IoU)

or Jaccard similarity / Jaccard index

$$= \frac{S_\cap}{S_\cup} \quad \frac{\text{intersection}}{\text{union}}$$

Pred v.s. ground-truth box (GT):

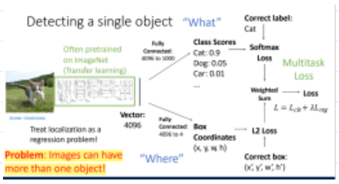
IoU > 0.5 decent

> 0.7 pretty good

> 0.9 almost perfect

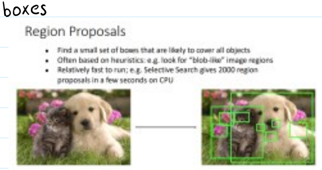
4 Models

1° To detect a single obj.

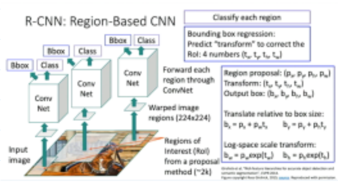


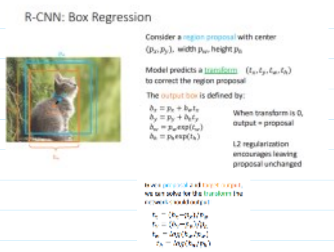
2° To det muliple obj

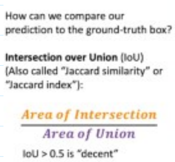
↓ # boxes

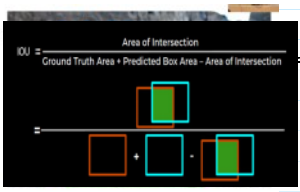
⇒ 

3° R-CNN (Region-CNN)









Pb1

⇒ Non-Max Suppression solve overlapping boxes

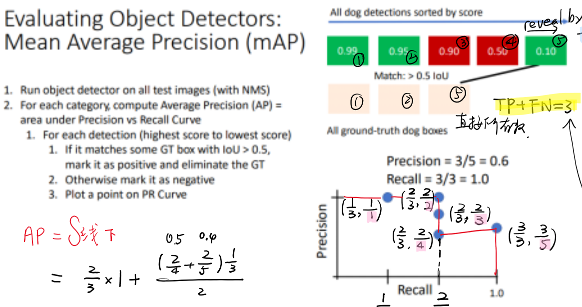
消除IoU大的(therehold), scoring 低的那个 直到无叠 > thereshold.

1. Select next highest scoring box
2. Eliminate lower-scoring boxes with IoU > threshold (e.g. 0.7)
3. If any boxes remain, GOTO 1

⊖ obj overlapping

Pb2: L2 regu encourages leaving proposal unchanged

Sol: mean Avg. Precision (mAp)



$AP = S_{线下}$

$= \frac{2}{3}\times 1 + \frac{(\frac{2}{4}+\frac{2}{5})\frac{1}{3}}{2}$

$= \frac{2}{3} + \frac{3}{20}$

$= \frac{40+9}{60} = \frac{49}{60} = 0.81$

Steps	precision	recall
①	$\frac{1}{1}$ (TP)	$\frac{1}{3}$ 3个GT的1个TP
②	$\frac{2}{2}$ (2TP)	$\frac{2}{3}$ 2
③	$\frac{2}{3}$ $(\frac{2TP}{2TP+1FP})$	〃
④	$\frac{2}{4}$ (2 2)	〃
⑤	$\frac{3}{5}$	$\frac{3}{3}$

下个分母+1; if TP, 分子+1.

分母一直为 TP+FN; if TP, 分子+1

Confusion matrix

		Actual +	Actual −
Predicted	+	TP	FP
	−	FN	TN

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

① data balanced
② easy to exp
③ 每类 label 一样重要

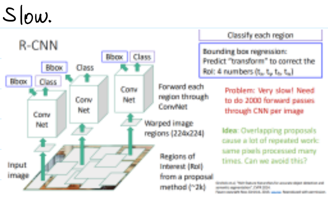
$precision = \frac{TP}{TP+FP}$

查准率: 在预测为T中, 有多少真T. 求精.
① 冤假错案成本高: 垃圾邮件.

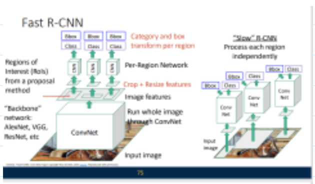
$recall = \frac{TP}{TP+FN}$

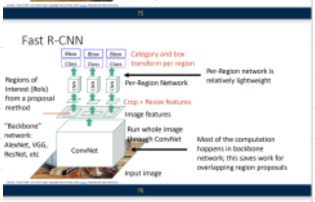
查全: 实际为T的中有多少被预测到.
① 漏网之鱼成本高: 地震报警.

Pb3 Slow.



4° Fast R-CNN





Crop features

1° RoI Pool

different resolution here

snap 折断

Cropping Features: RoI Pool

: RoI Pool "Snap" to

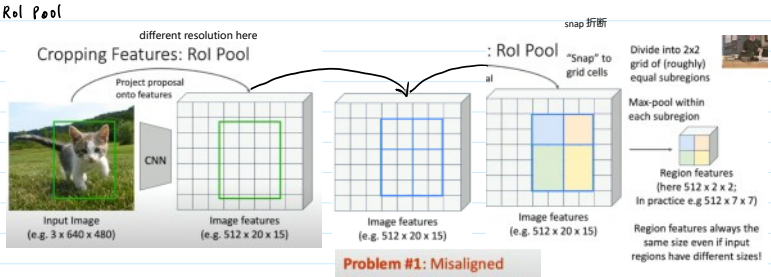
Divide into 2x2

Crop features

1° RoI Pool

different resolution here

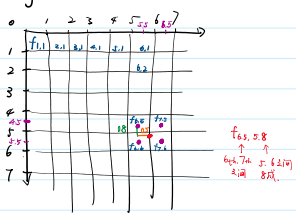
snap 折断



Problem #1: Misaligned features due to snapping
Problem #2: Can't backprop to box coordinates

2° RoI Align

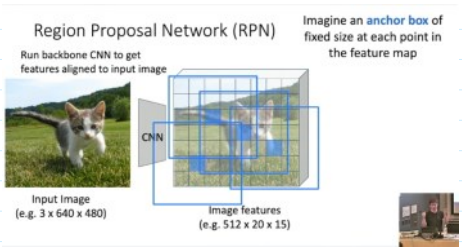
bilinear interpolation



feat $f_{6.5,7.8} \times (1-0.8) \times (1-0.5)$

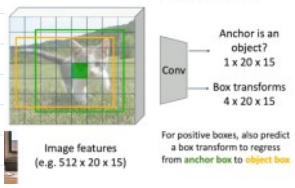
$$f_{6.5,7.8} = (f_{6,7} * 0.5 * 0.2) + (f_{7,7} * 0.5 * 0.2) + (f_{6,8} * 0.5 * 0.8) + (f_{7,8} * 0.5 * 0.8)$$

can be backpropagate to box coordinates!

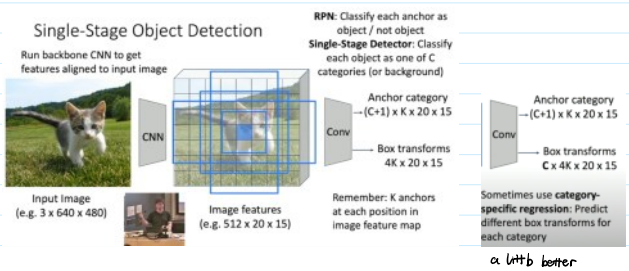


Train a cnn: classify anchor boxes as either containing an obj or not.

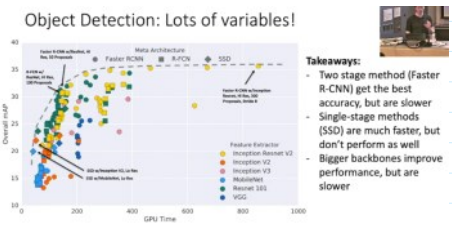
\+ −

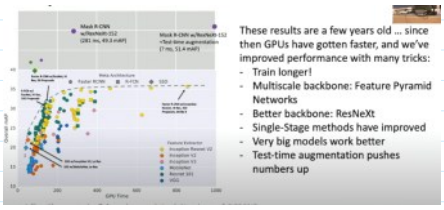


Problem: Anchor box may have the wrong size / shape
Solution: Use **K** different anchor boxes at each point!



a little better





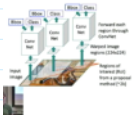
These results are a few years old ... since then GPUs have gotten faster, and we've improved performance with many tricks:

- Train longer!
- Multiscale backbone: Feature Pyramid Networks
- Better backbone: ResNeXt
- Single-Stage methods have improved
- Very big models work better
- Test-time augmentation pushes numbers up

Object Detection Methods

Both of these rely on anchor boxes. Can we do detection without anchors?

"Slow" R-CNN: Run CNN independently for each region



Fast R-CNN: Apply differentiable cropping to shared image features

Faster R-CNN: Compute proposals with CNN

Single-Stage: Fully convolutional detector

Yes cornerNet