Object Detection

KOHI 2022

2022.09.17.

University of Ulsan, Asan Medical Center

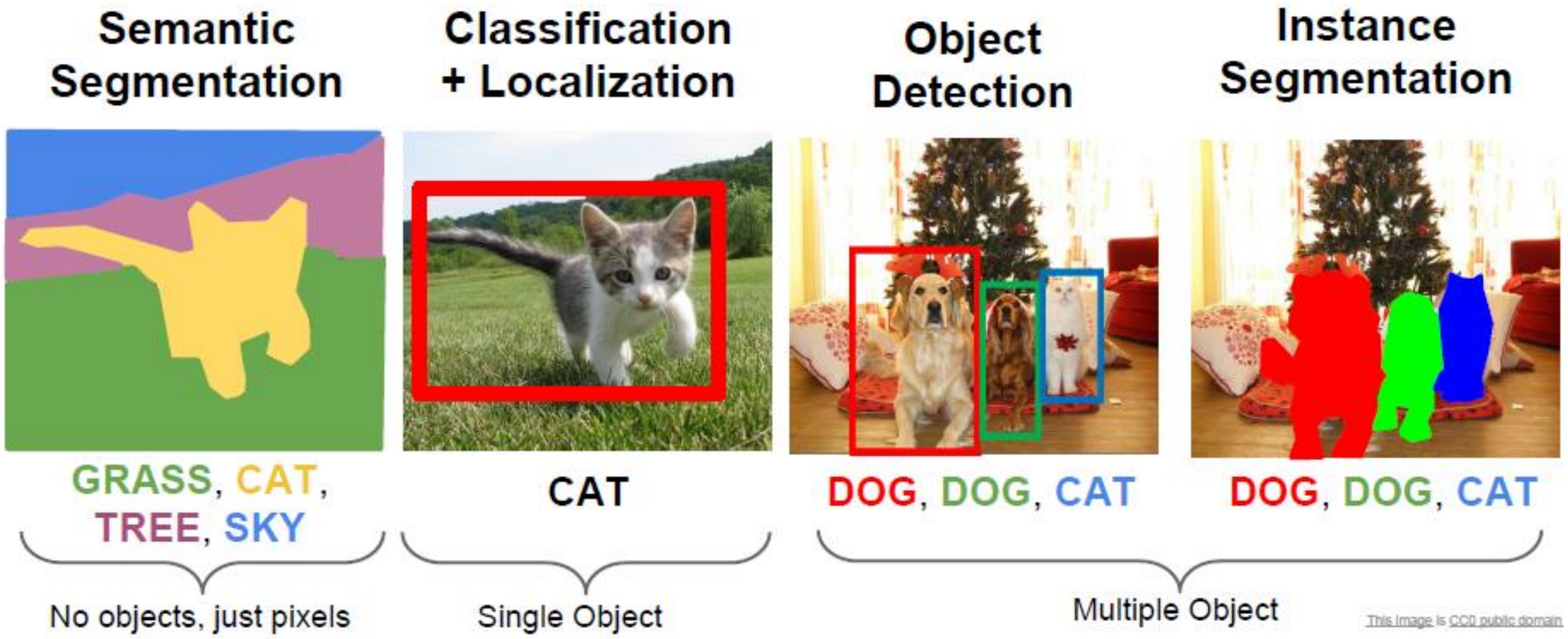
Keewon Shin PhDc

Contents

- 1. **Computer vision tasks**
  - Object detection
  - Instance segmentation
  - Evaluation
  - Trend
  - Conclusion

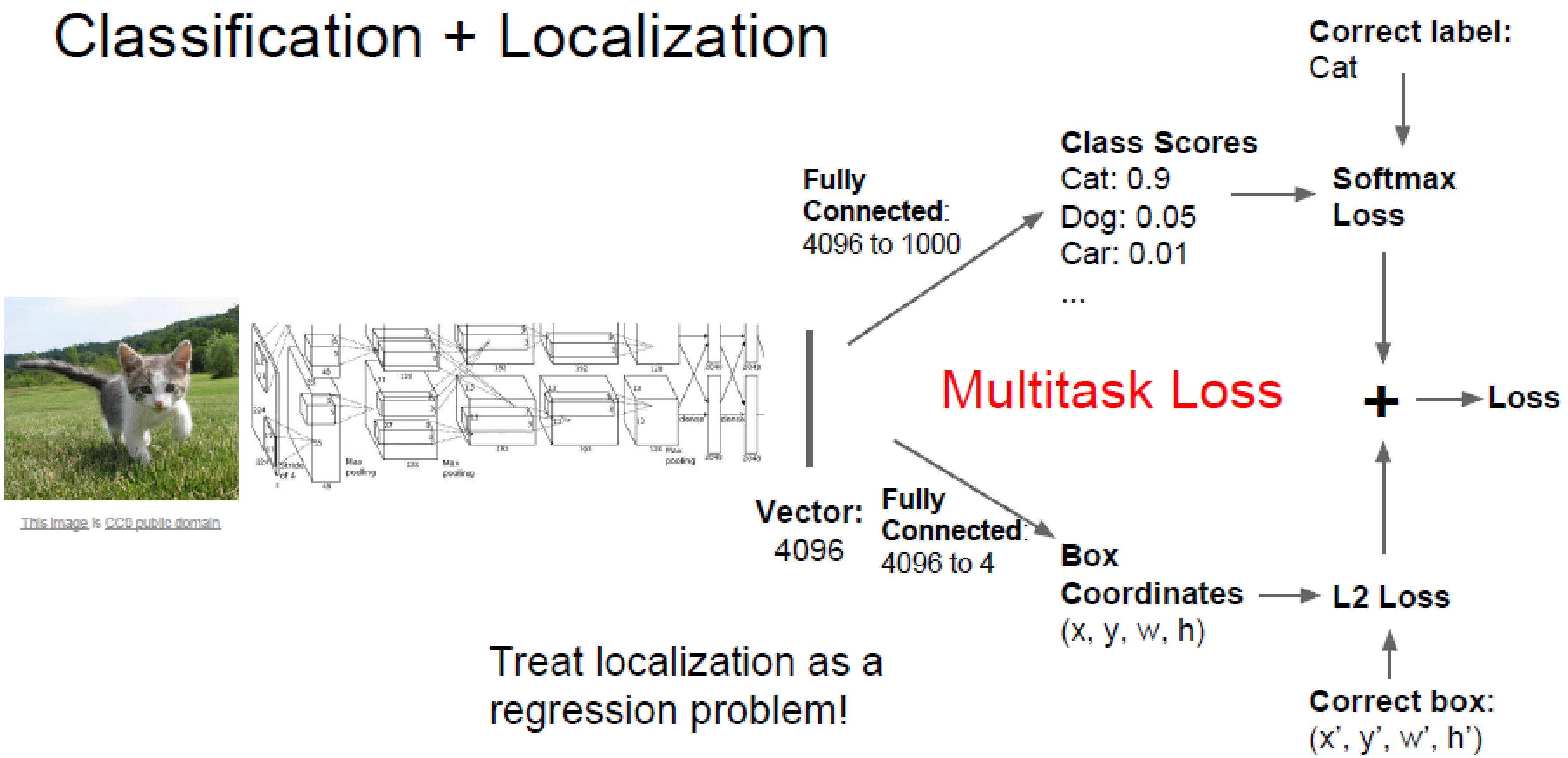
- 2. **Hands on : Mask-RCNN**
  (Instance segmentation and evaluation)

Computer Vision Tasks



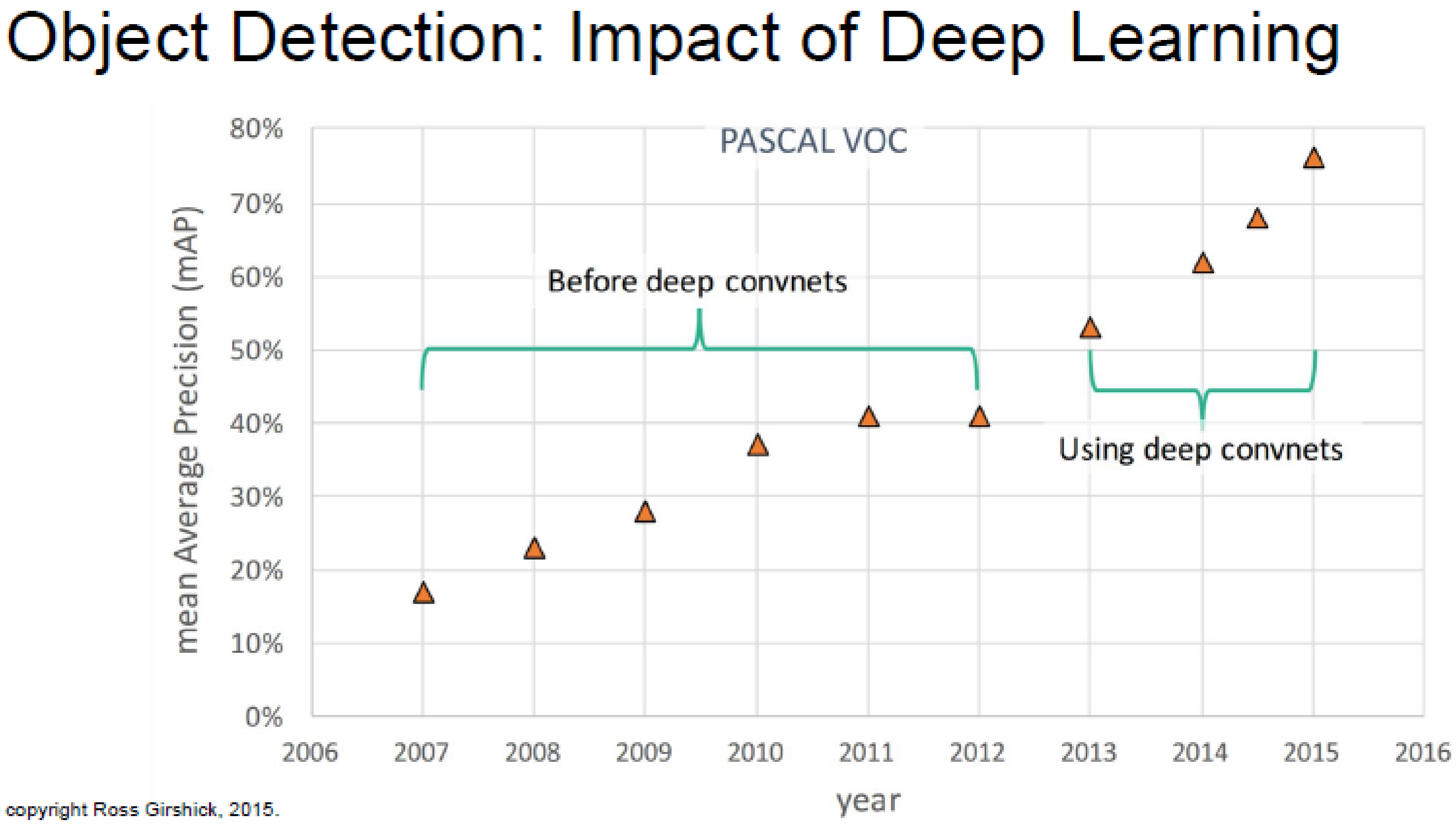
Lecture note from CS231N, http://cs231n.stanford.edu/

Classification + Localization

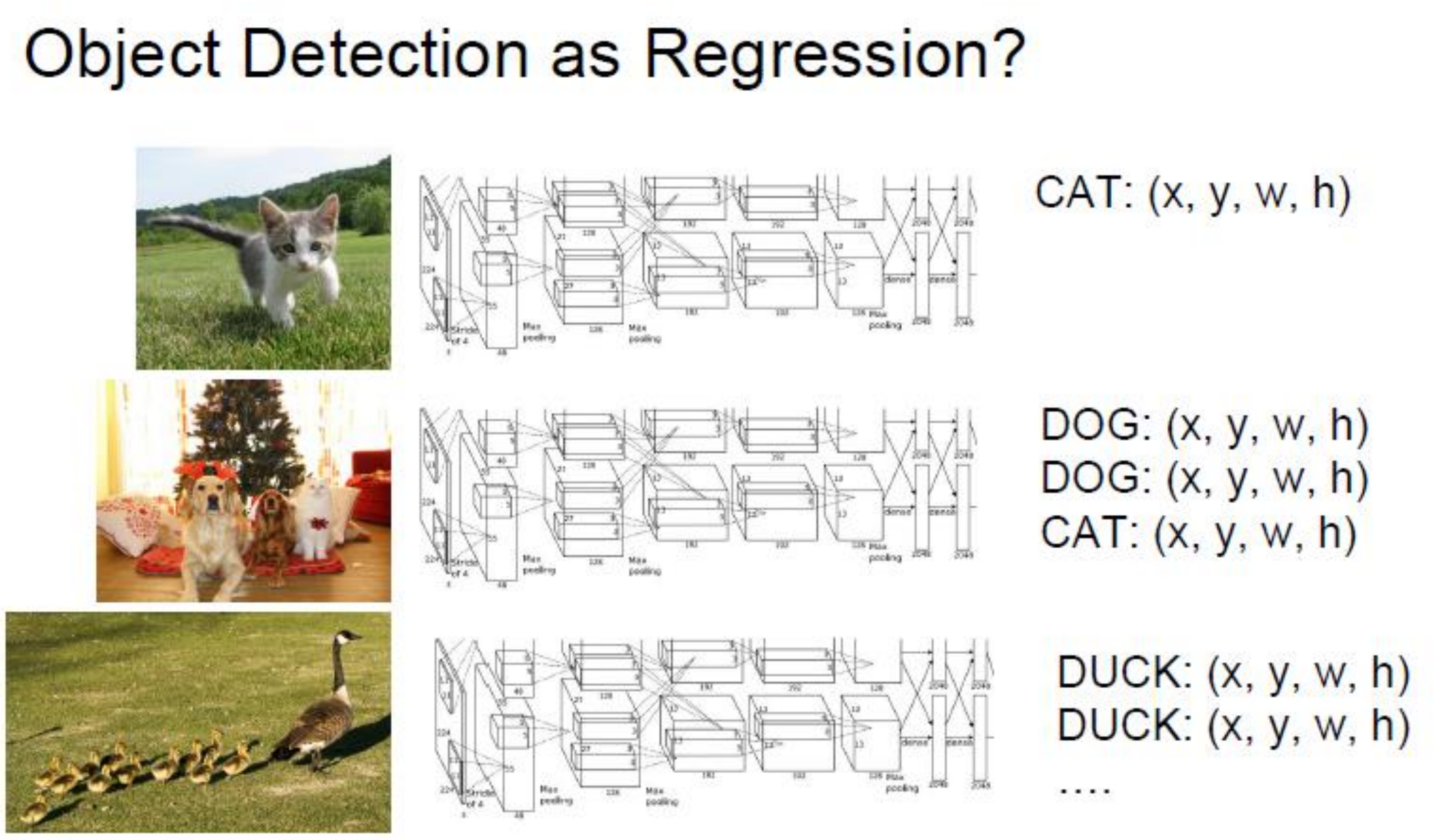


참고 : (x,y,w,h) or (x1,y1,x2,y2), Box를 정의하는 4개의 숫자면 됨

Object Detection

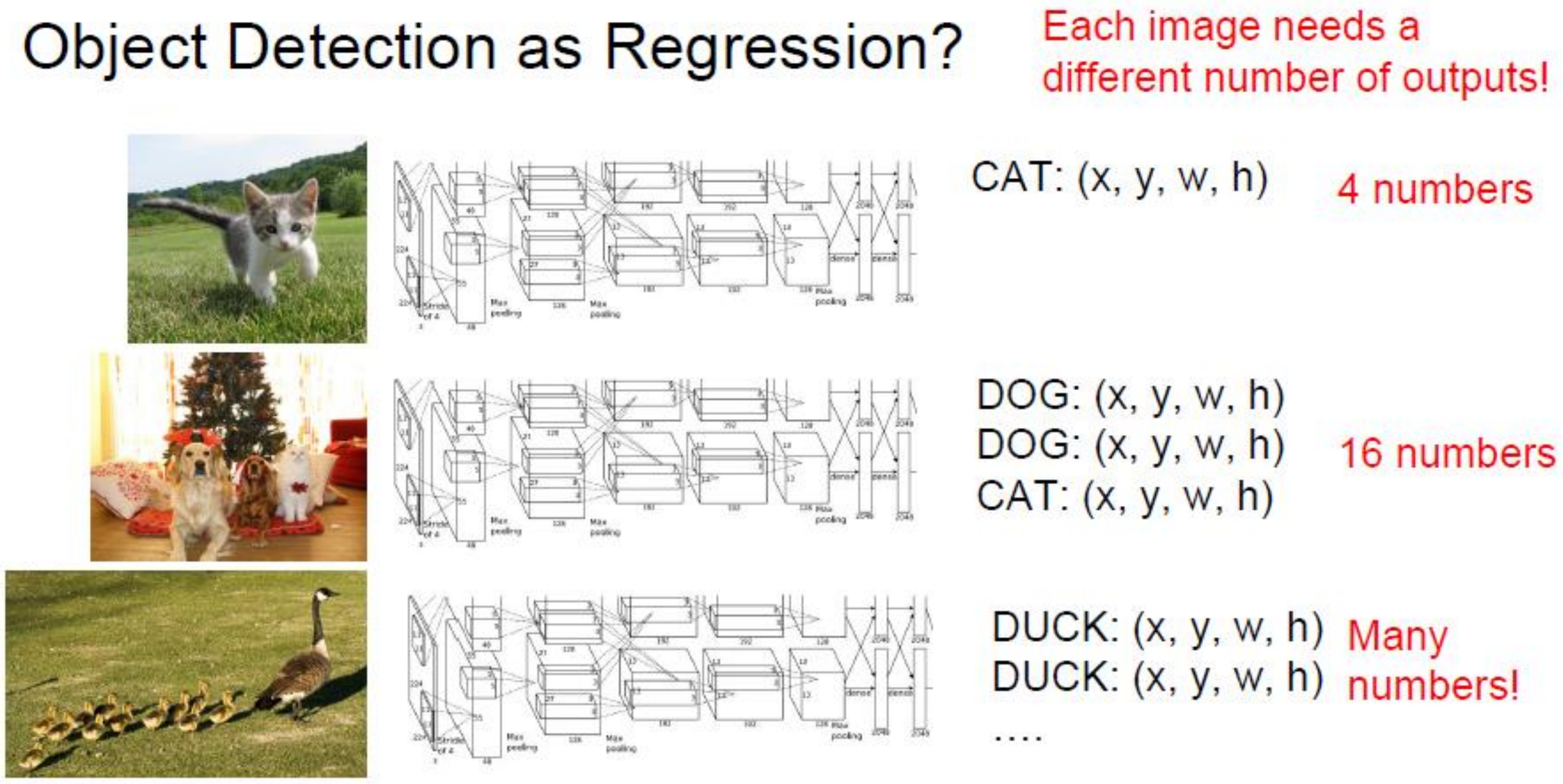


Object Detection



Lecture note from CS231N, http://cs231n.stanford.edu/

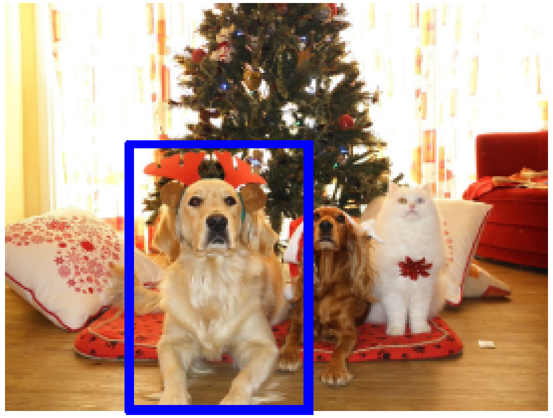
Object Detection



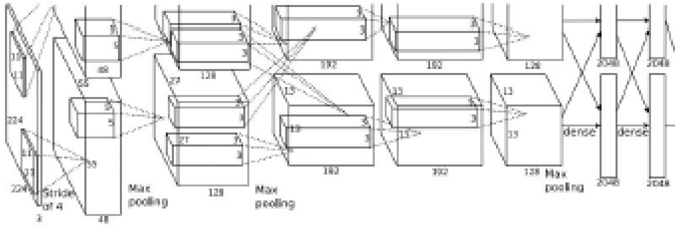
Lecture note from CS231N, http://cs231n.stanford.edu/

Object Detection

Object Detection as Classification: Sliding Window



Apply a CNN to many different crops of the image, CNN classifies each crop as object or background



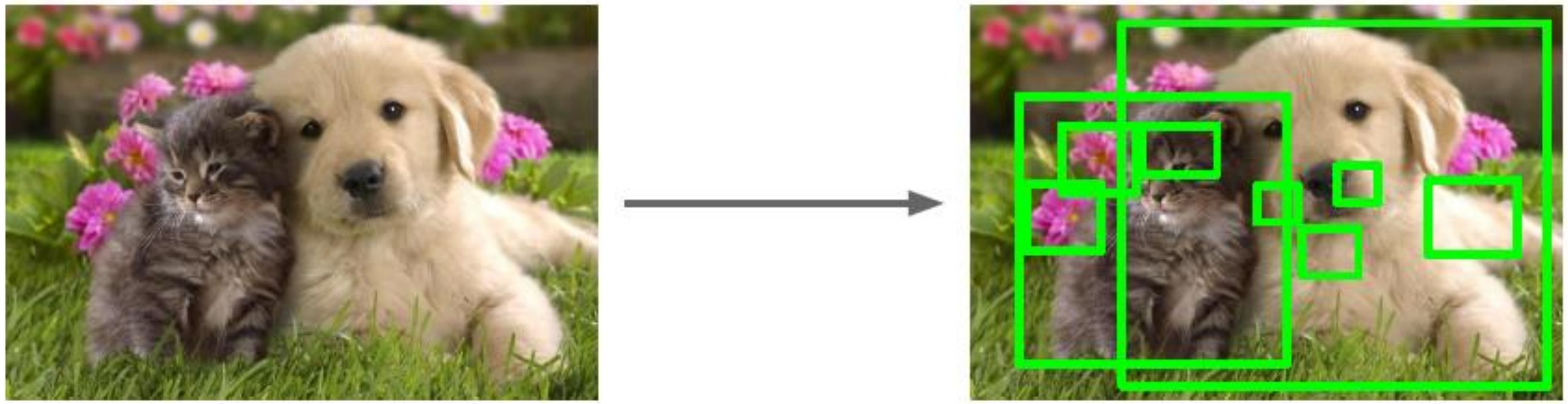
Dog? YES
Cat? NO
Background? NO

Lecture note from CS231N, http://cs231n.stanford.edu/

Object Detection

Region Proposals

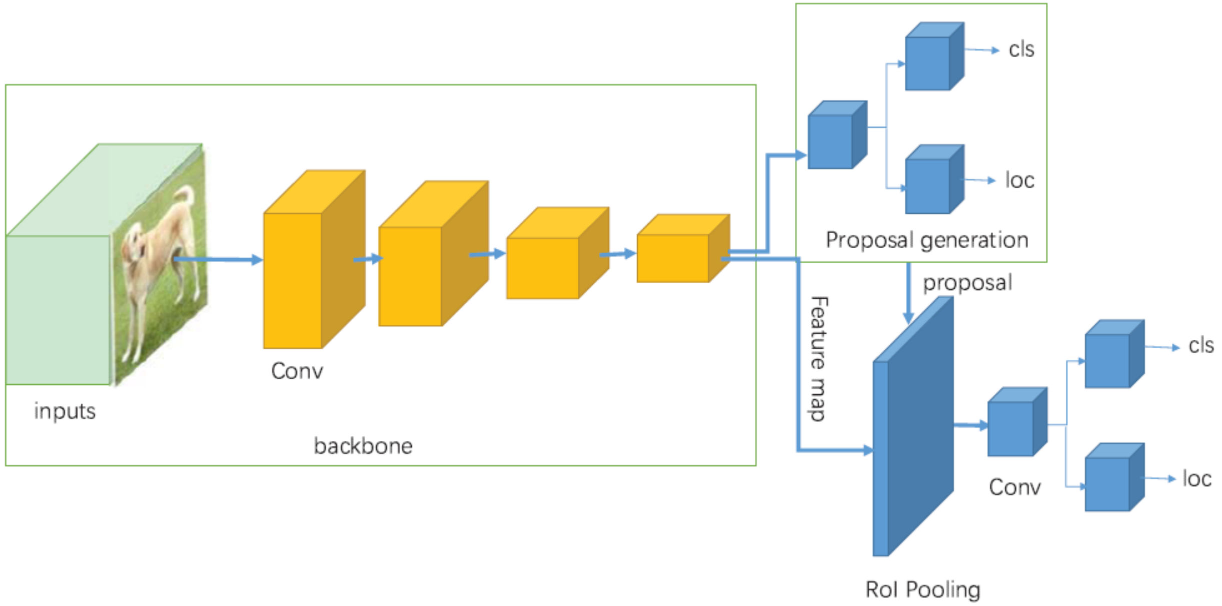
- Find “blobby” image regions that are likely to contain objects
- Relatively fast to run; e.g. Selective Search gives 1000 region proposals in a few seconds on CPU



Alexe et al, “Measuring the objectness of image windows”, TPAMI 2012
Uijlings et al, “Selective Search for Object Recognition”, IJCV 2013
Cheng et al, “BING: Binarized normed gradients for objectness estimation at 300fps”, CVPR 2014
Zitnick and Dollar, “Edge boxes: Locating object proposals from edges”, ECCV 2014

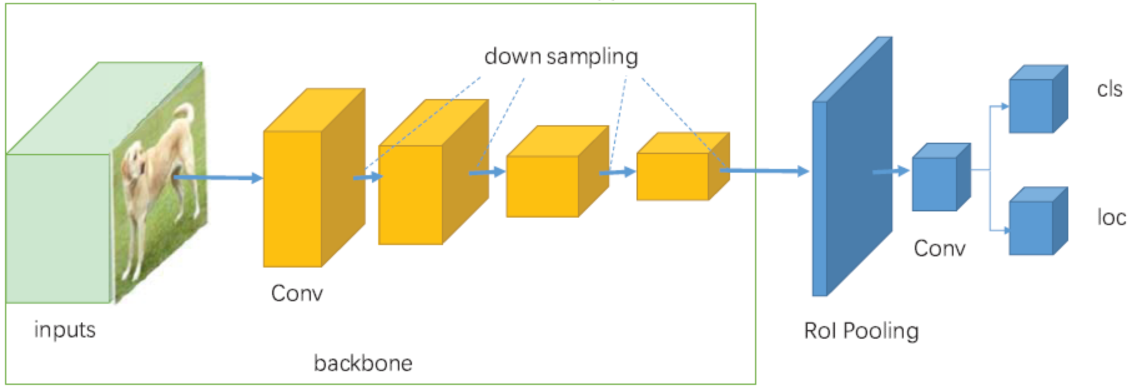
Object Detection

2 Stage approach



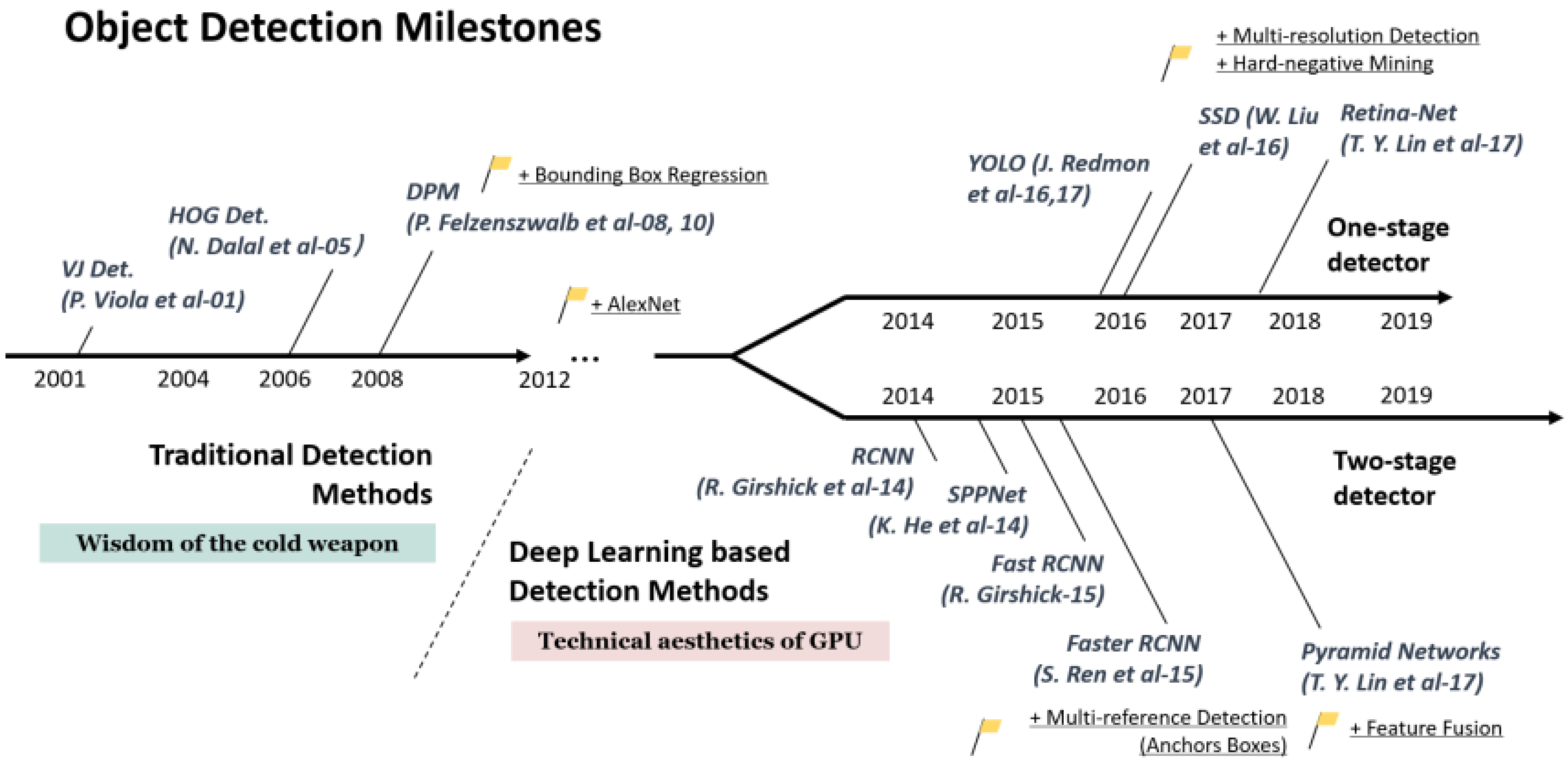
(a)

1 Stage approach



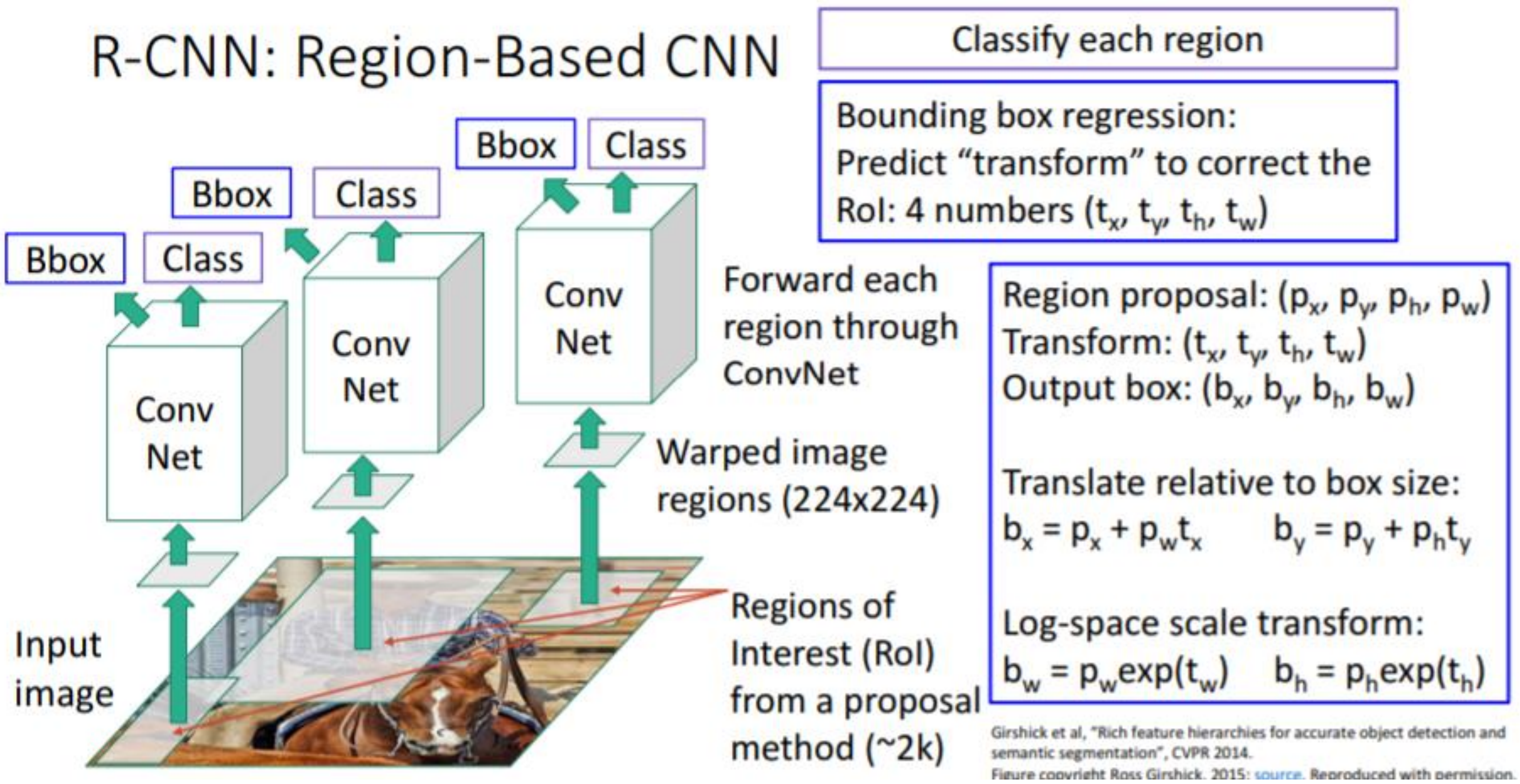
(b)

Object Detection



https://chacha95.github.io/2020-02-26-Object-Detection3/

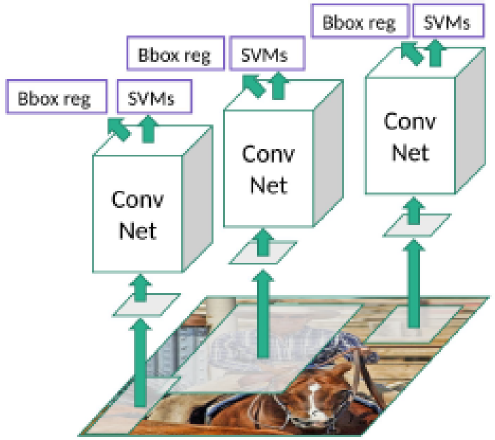
Object Detection



Object Detection

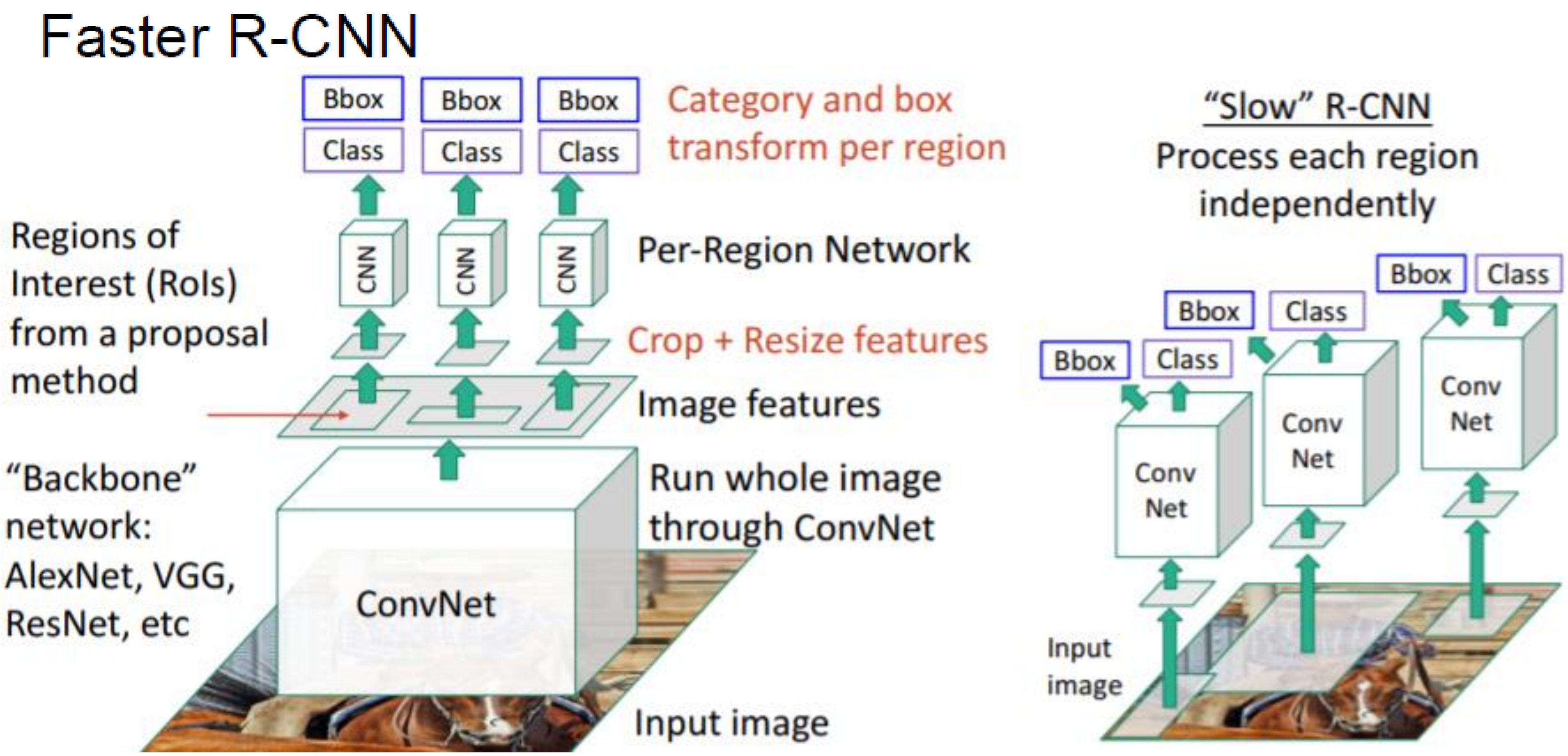
R-CNN: Problems

- Ad hoc training objectives
  - Fine-tune network with softmax classifier (log loss)
  - Train post-hoc linear SVMs (hinge loss)
  - Train post-hoc bounding-box regressions (least squares)
- Training is slow (84h), takes a lot of disk space
- Inference (detection) is slow
  - 47s / image with VGG16 [Simonyan & Zisserman. ICLR15]
  - Fixed by SPP-net [He et al. ECCV14]



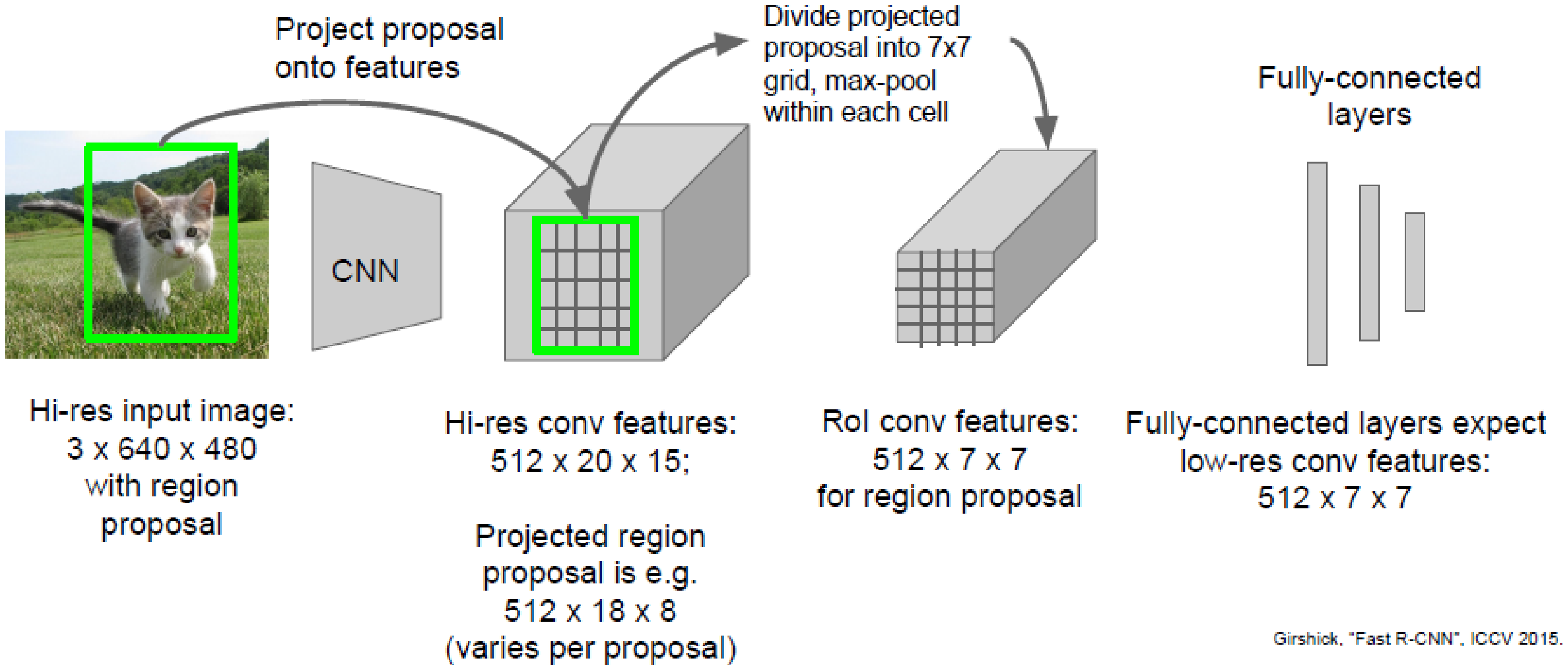
Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.

Object Detection



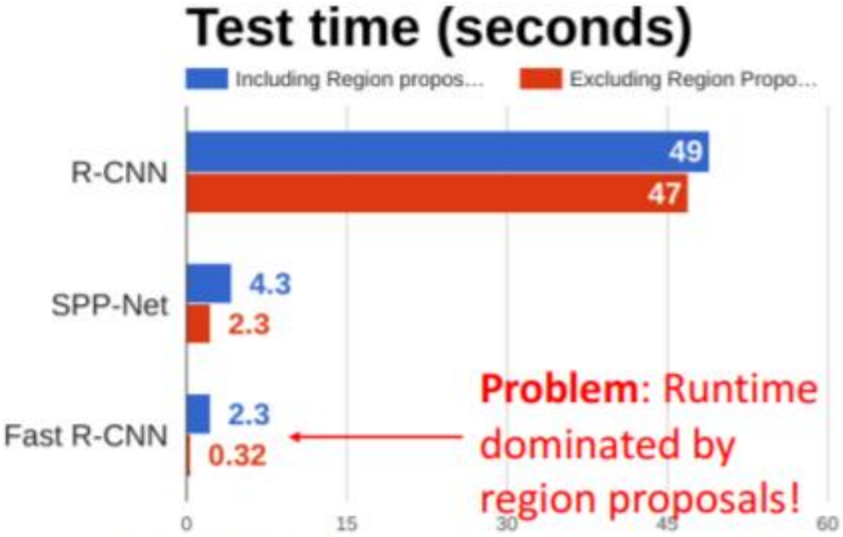
Object Detection

Faster R-CNN: RoI Pooling



Object Detection

R-CNN vs SPP vs Fast R-CNN



Problem: Runtime dominated by region proposals!

Recall: Region proposals computed by heuristic "Selective Search" algorithm on CPU -- let's learn them with a CNN instead!

Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
He et al, "Spatial pyramid pooling in deep convolutional networks for visual recognition", ECCV 2014
Girshick, "Fast R-CNN", ICCV 2015

Object Detection

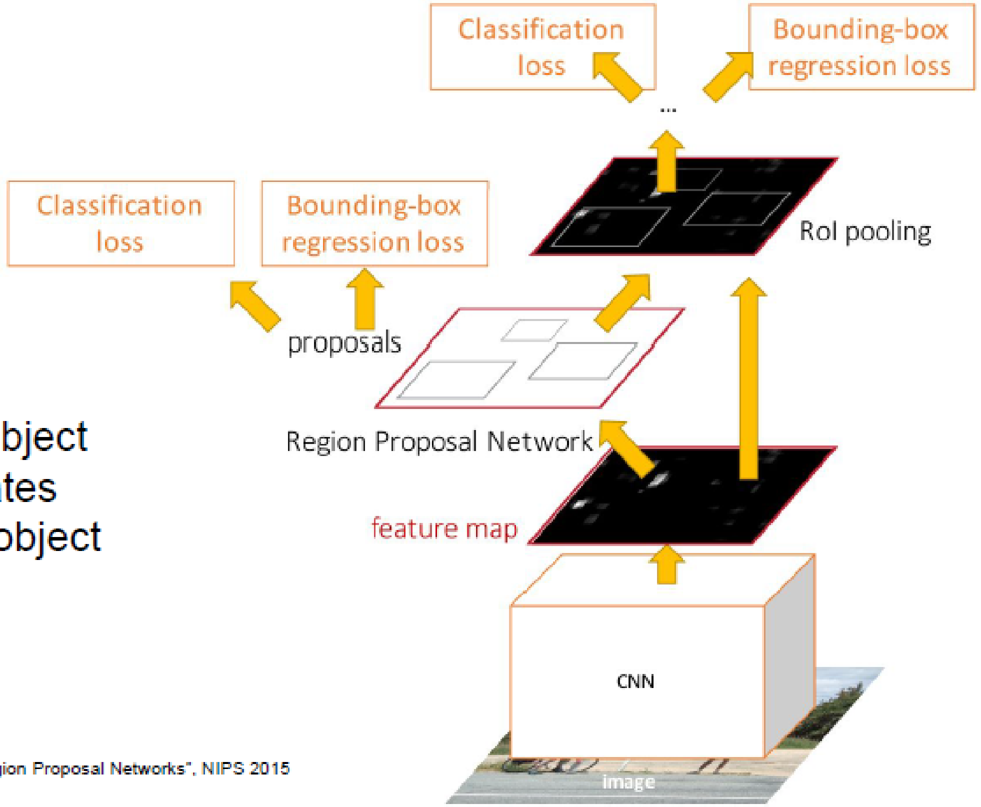
Faster R-CNN:

Make CNN do proposals!

Insert **Region Proposal Network (RPN)** to predict proposals from features

Jointly train with 4 losses:

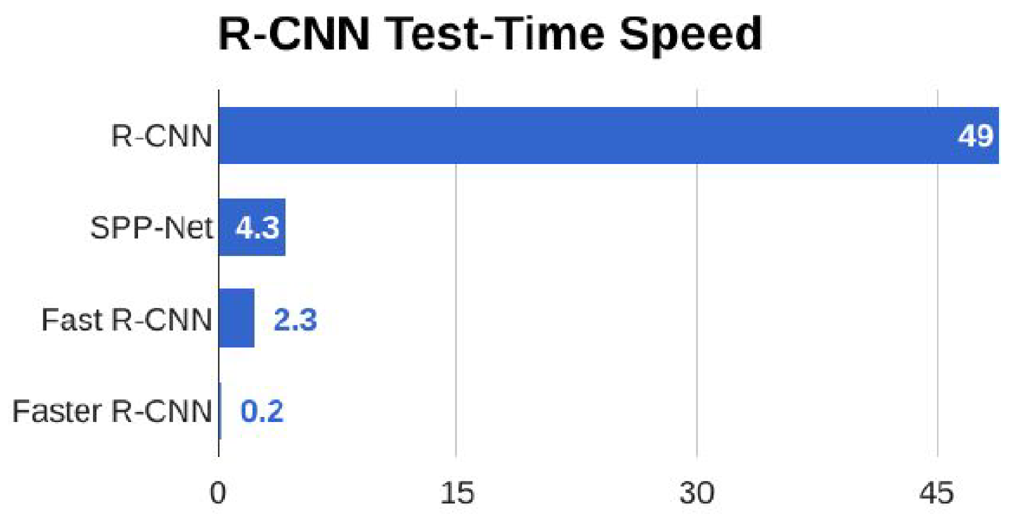
1. RPN classify object / not object
2. RPN regress box coordinates
3. Final classification score (object classes)
4. Final box coordinates



Ren et al, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NIPS 2015

Object Detection

Faster R-CNN:
Make CNN do proposals!

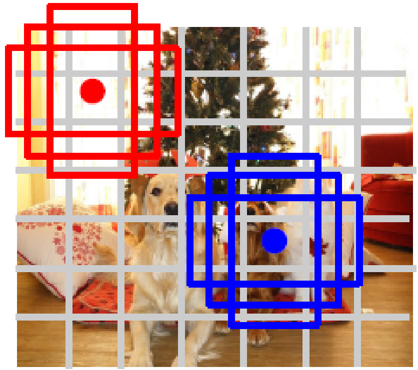


Object Detection

Detection without Proposals: YOLO / SSD

Go from input image to tensor of scores with one big convolutional network!

Input image
3 x H x W



Divide image into grid
7 x 7

Image a set of **base boxes** centered at each grid cell
Here B = 3

Within each grid cell:
- Regress from each of the B base boxes to a final box with 5 numbers: (dx, dy, dh, dw, confidence)
- Predict scores for each of C classes (including background as a class)

Output:
7 x 7 x (5 * B + C)

Redmon et al, "You Only Look Once: Unified, Real-Time Object Detection", CVPR 2016
Liu et al, "SSD: Single-Shot MultiBox Detector", ECCV 2016

Object Detection : summary

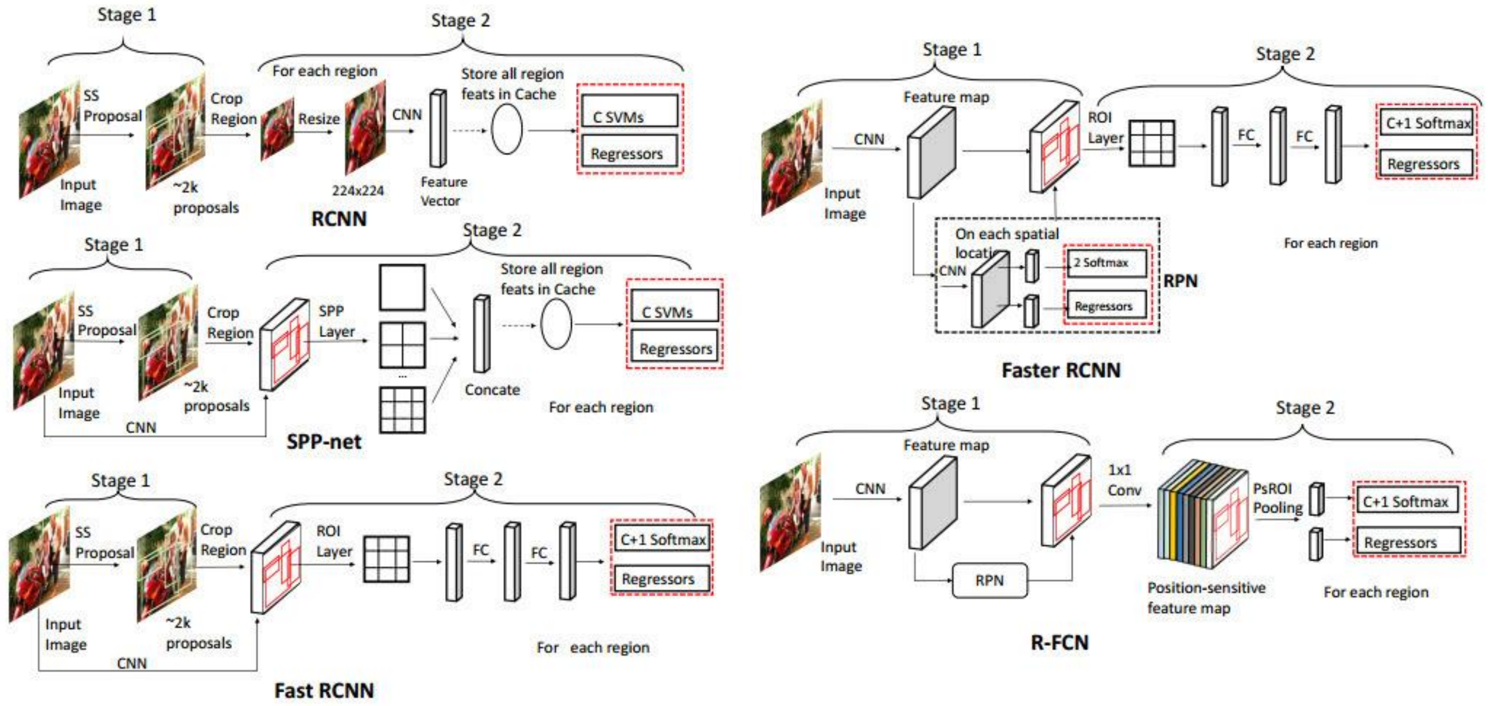


Figure 4: Overview of different two-stage detection frameworks for generic object detection. Red dotted rectangles denote the outputs that define the loss functions.

Object Detection : summary

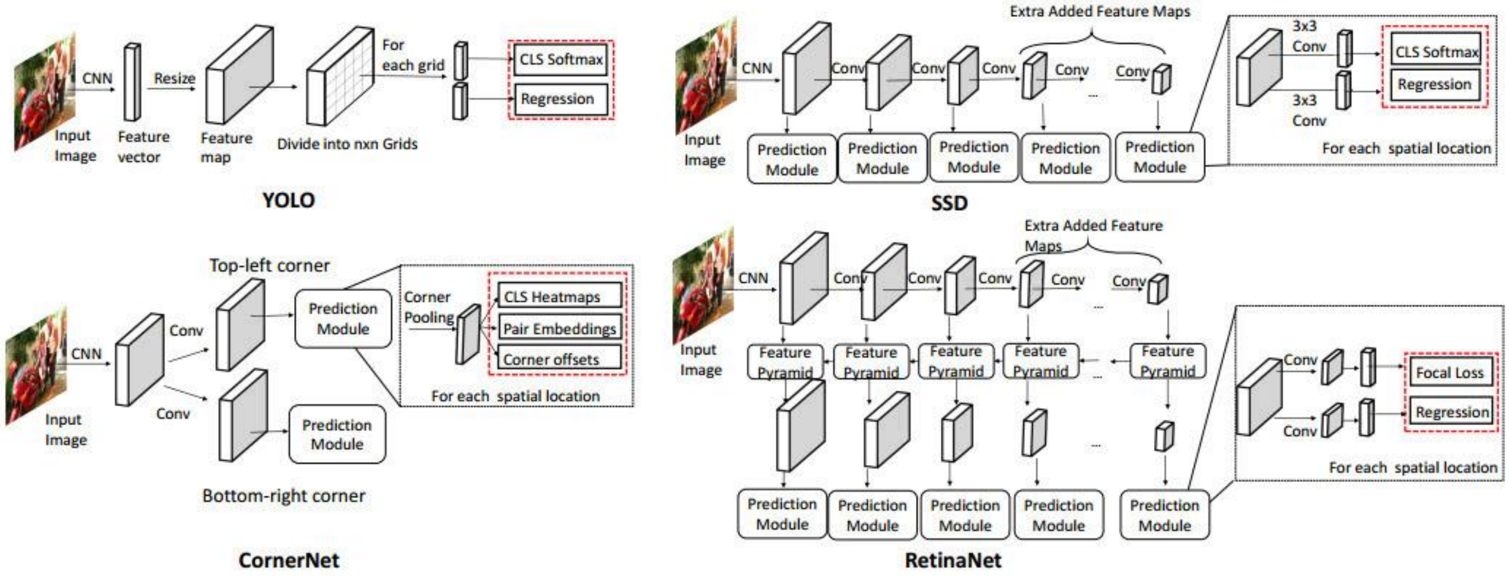
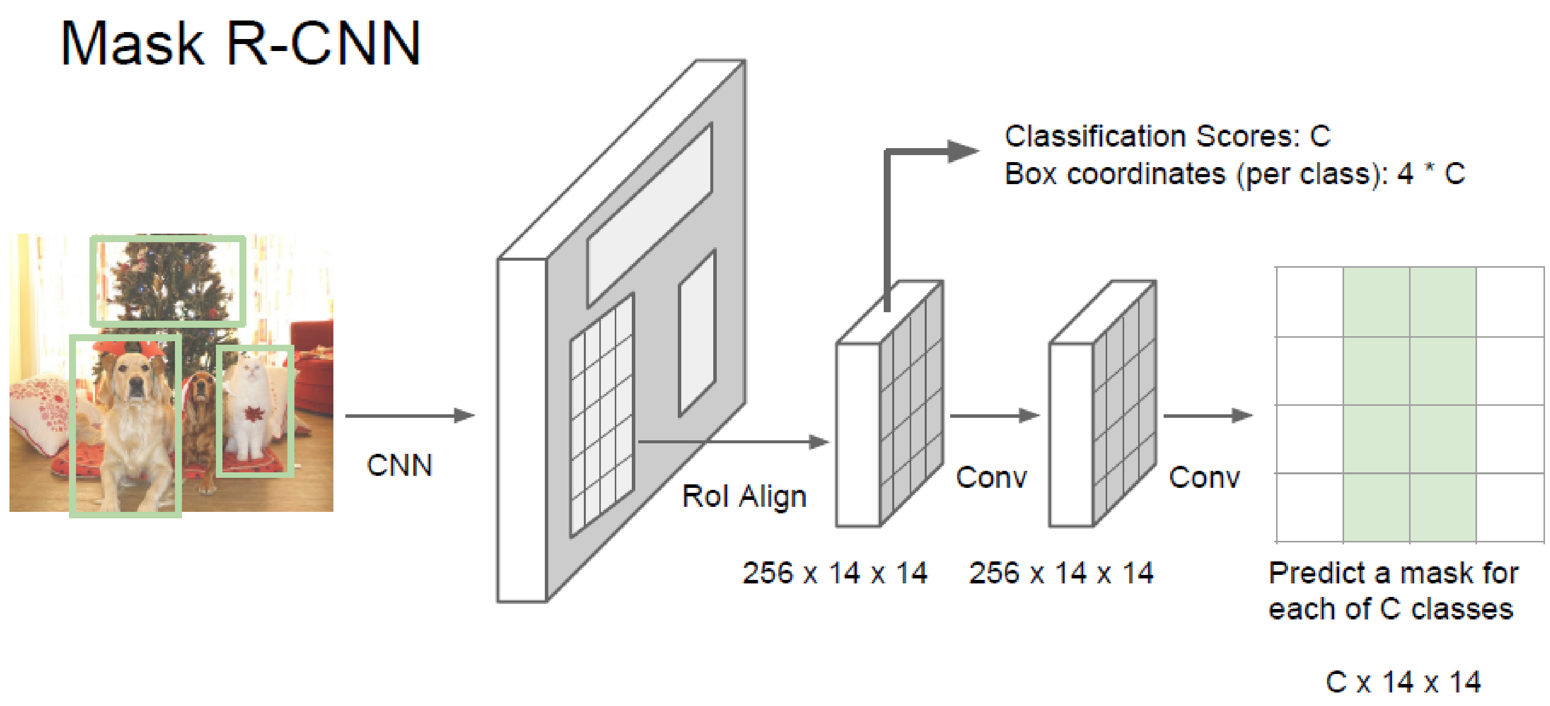


Figure 5: Overview of different one-stage detection frameworks for generic object detection. Red rectangles denotes the outputs that define the objective functions.

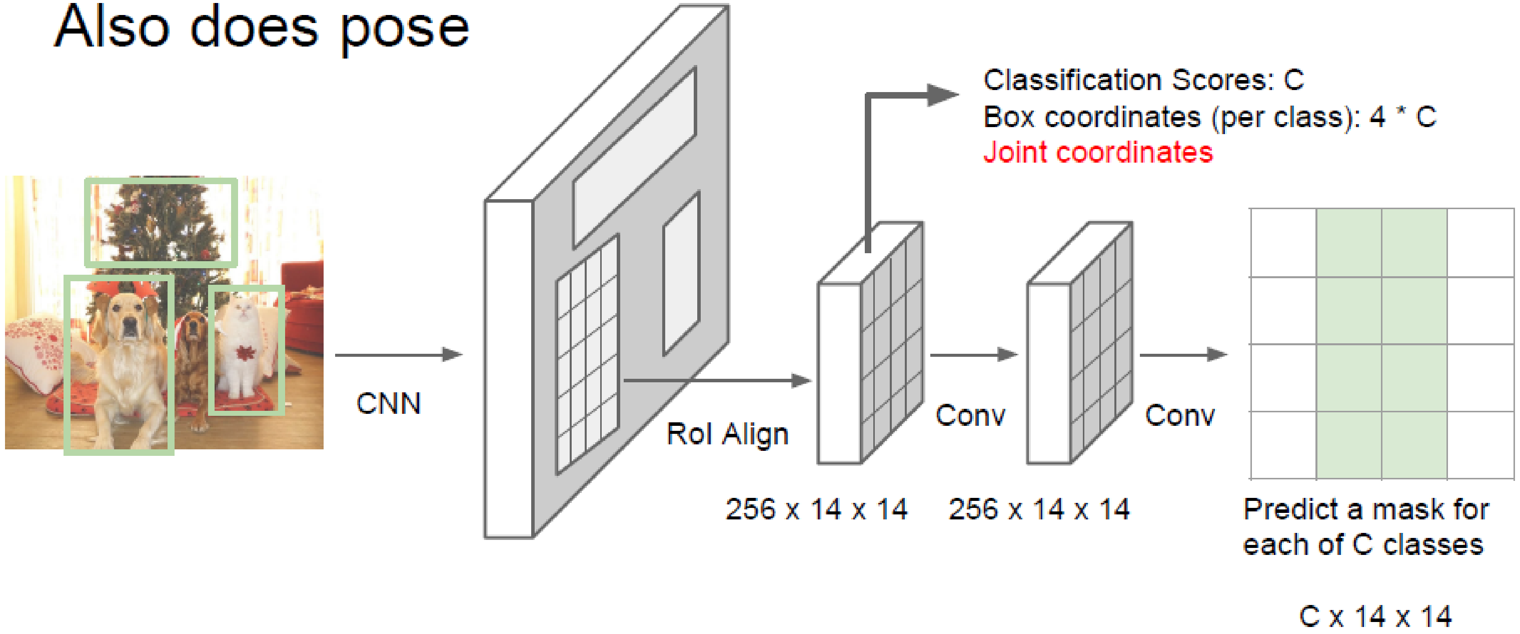
Instance segmentation



He et al, "Mask R-CNN", arXiv 2017

Instance segmentation

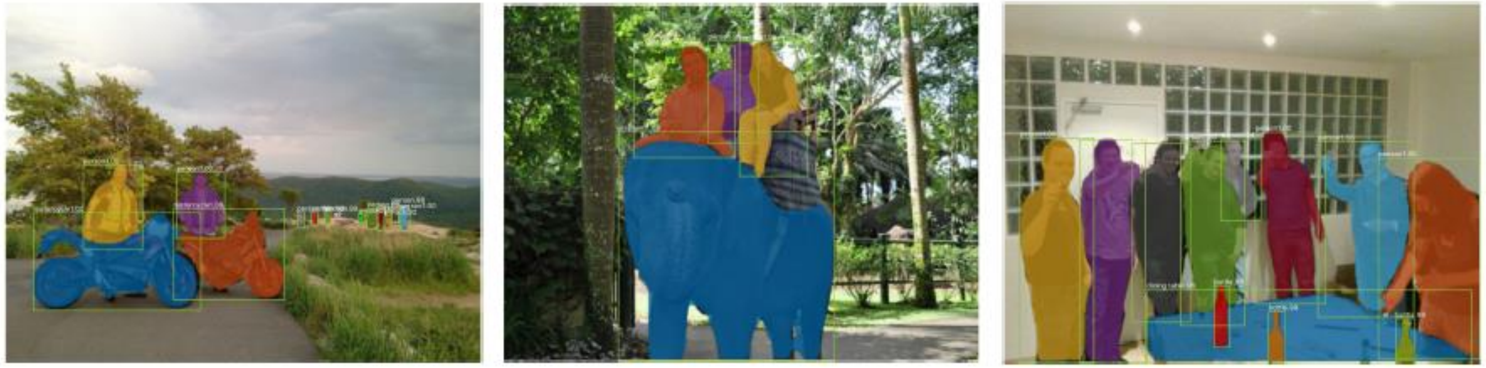
Mask R-CNN
Also does pose



He et al. "Mask R-CNN". arXiv 2017

Instance segmentation

Mask R-CNN: Very Good Results!



He et al, "Mask R-CNN", arXiv 2017

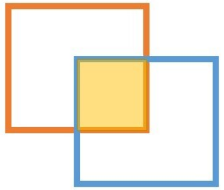
Instance segmentation

Mask R-CNN
Also does pose

He et al, "Mask R-CNN", arXiv 2017
Figures copyright Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick, 2017.
Reproduced with permission.

Evaluation

- IoU(Intersection over union)



$$Intersection\ over\ Union\ (IoU) = \frac{Area\ of\ Overlap}{Area\ of\ Union}$$

- Precision and Recall

$$Precision = \frac{TP}{TP+FP} \quad Recall = \frac{TP}{TP+FN} \quad F1score = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

where,
TP be the number of true positives,
FP be the number of false positives,
FN be the number of false negatives

		Predicted Class	
		Negative	Positive
Actual Class	Negative	TN (True Negative) 음성으로 예측했고 실제 범주는 음성인 경우의 수	FP (False Positive) 양성으로 예측했지만 실제 범주는 음성인 경우의 수
	Positive	FN (False Negative) 음성으로 예측했지만 실제 범주는 양성인 경우의 수	TP (True Positive) 양성으로 예측했고 실제 범주가 양성인 경우의 수

TN
Negative로 예측했고, True인 경우

FP
Positive로 예측했고, False인 경우

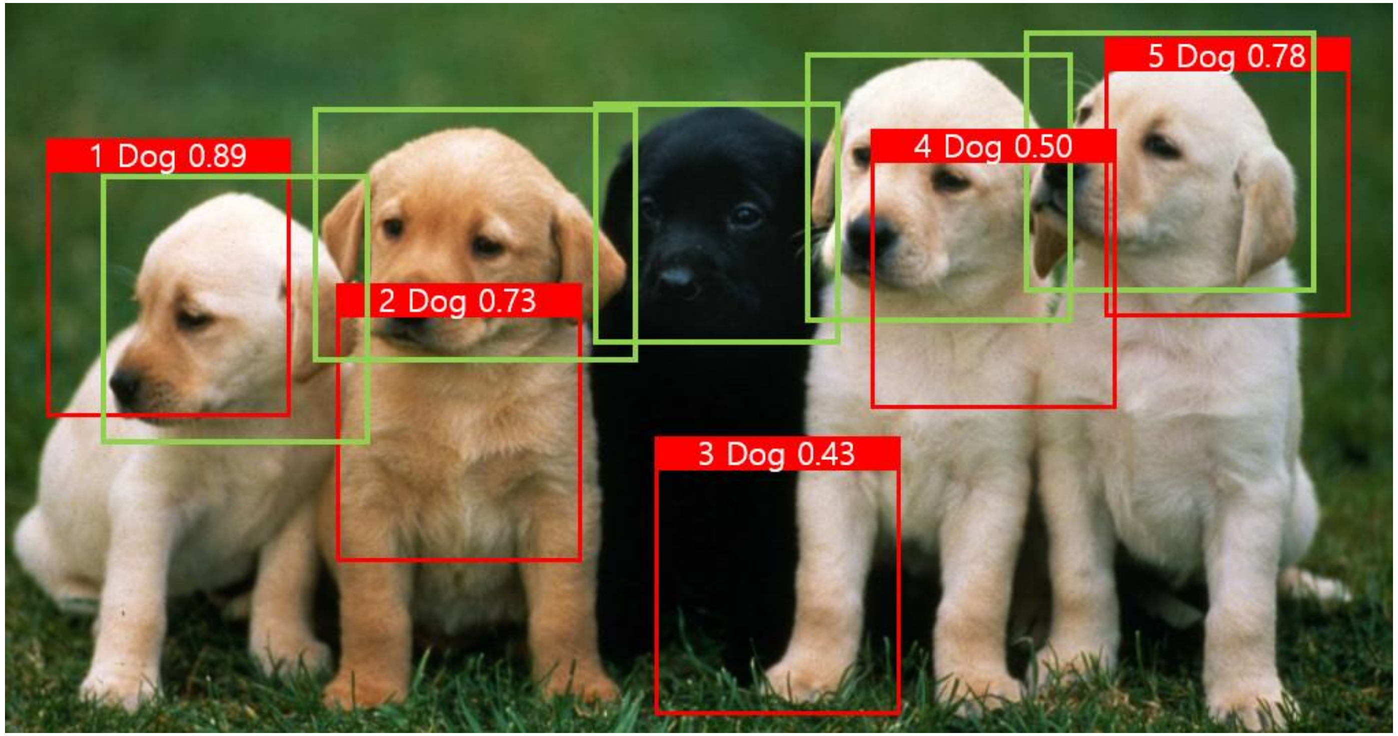
FN
Negative로 예측했고, False인 경우

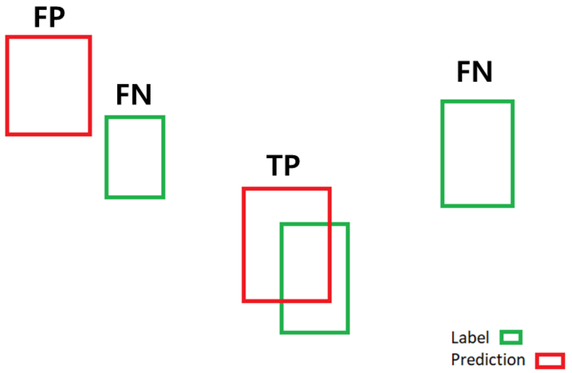
TP
Positive로 예측했고, True인 경우

https://herbwood.tistory.com/2

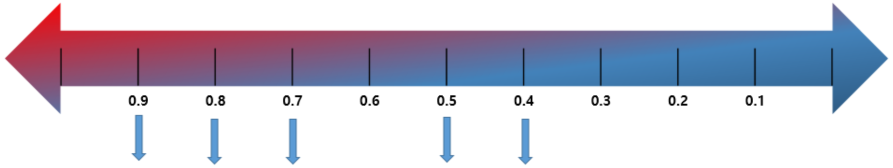
Evaluation

- Confidence score, distance





Confidence score Threshold



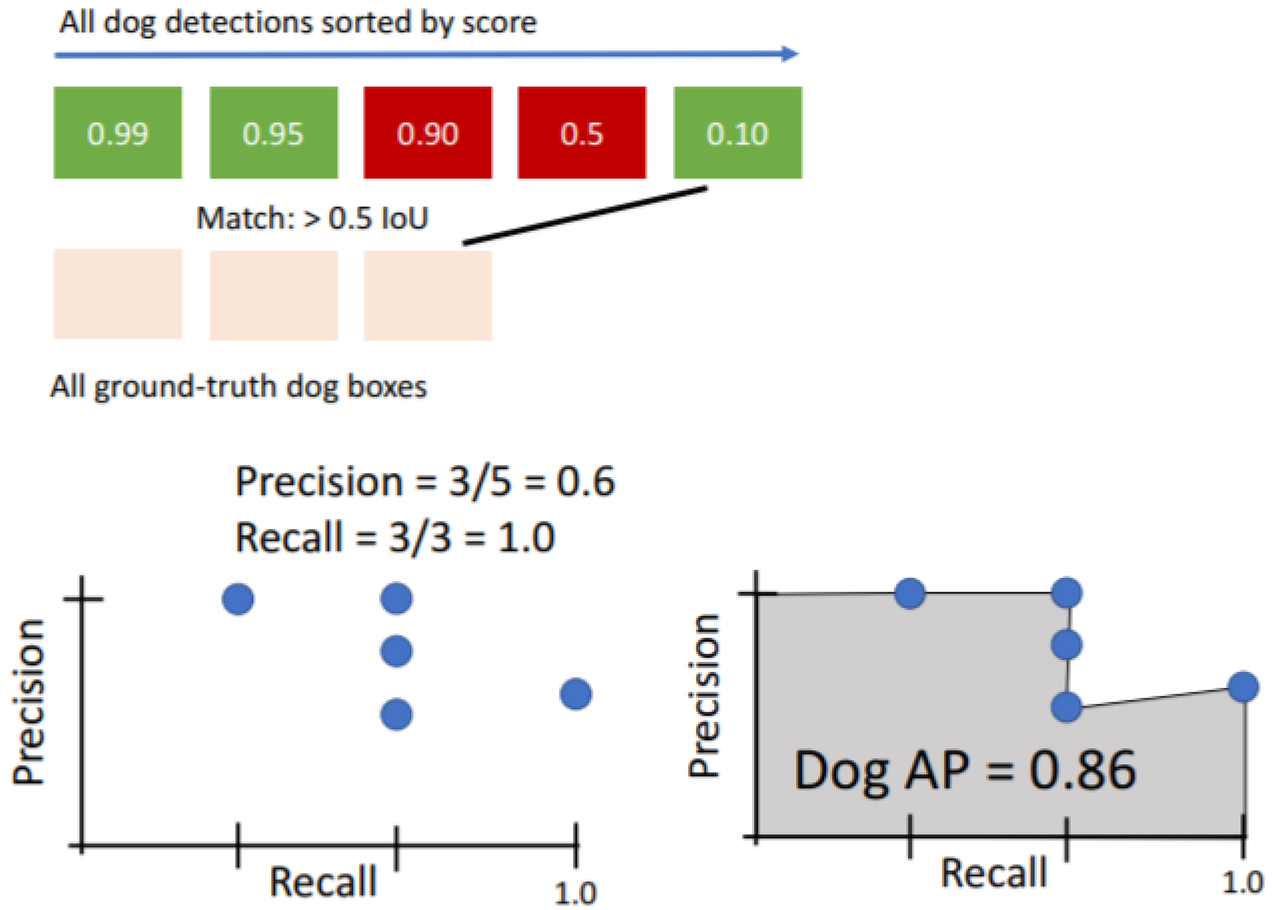
Bounding box	0개	1개	3개	4개	5개
Box index	X	1	1,2,5	1,2,4,5	1,2,3,4,5
Precision	0	1/1=1	3/3=1	¾=0.75	3/5=0.6
Recall	0	1/5=0.2	3/5=0.6	5/4=0.8	5/5=1

https://herbwood.tistory.com/2

Evaluation

- Evaluating Object Detectors: Mean Average Precision (mAP)

1. Run object detector on all test images (with NMS)
2. For each category, compute Average Precision (AP) = area under Precision vs Recall Curve
    1. For each detection (highest score to lowest score)
        1. If it matches some GT box with IoU > 0.5, mark it as positive and eliminate the GT
        2. Otherwise mark it as negative
        3. Plot a point on PR Curve



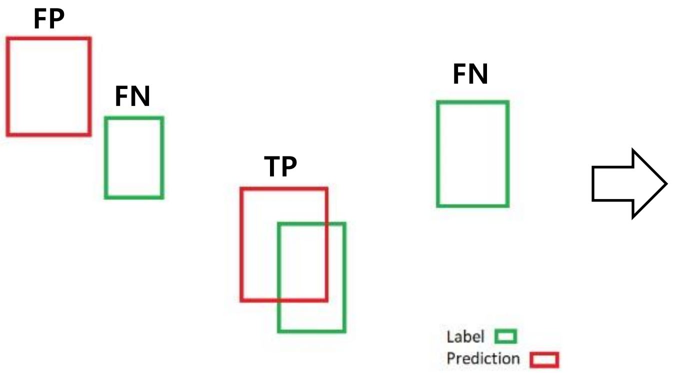
https://herbwood.tistory.com/2

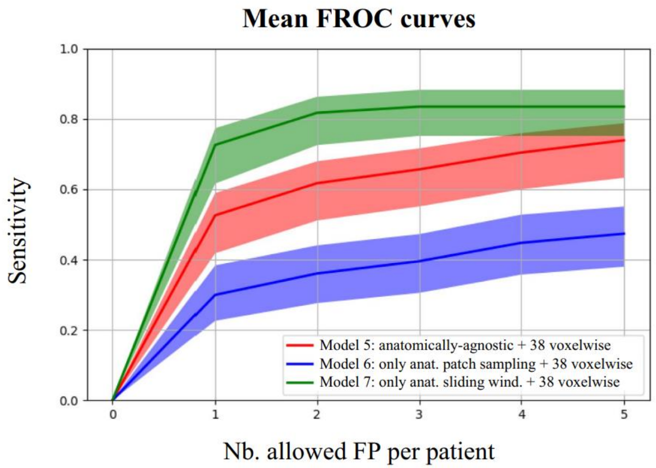
Evaluation

- Free-response receiver operating characteristic (FROC)

1. Slide-based Evaluation: The merits of the algorithms will be assessed for discriminating between slides containing metastasis and normal slides. Receiver operating characteristic (ROC) analysis at the slide level will be performed and the measure used for comparing the algorithms will be the area under the ROC curve (AUC).

2. **Lesion-based Evaluation**: For the lesion-based evaluation, free-response receiver operating characteristic (FROC) curve will be used. The FROC curve is defined as the plot of sensitivity versus the average number of false-positives per image.

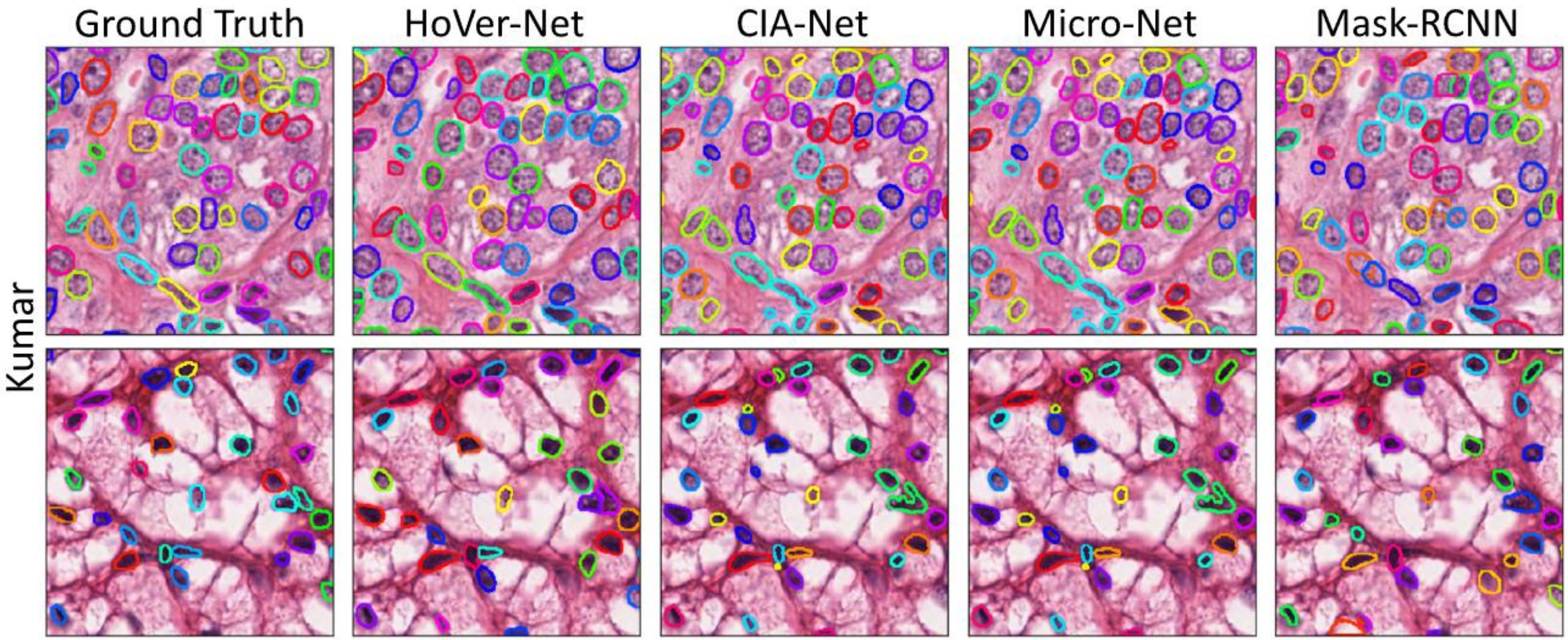




	A	B	C
1	Confidence	X coordinate	Y coordinate
2	0.73	18298	169828
3	0.84	10498	165754
4	0.57	12122	153638
5	0.91	10866	154596
6	0.32	13742	121722
7	0.21	12458	134585
8	0.64	14250	146531

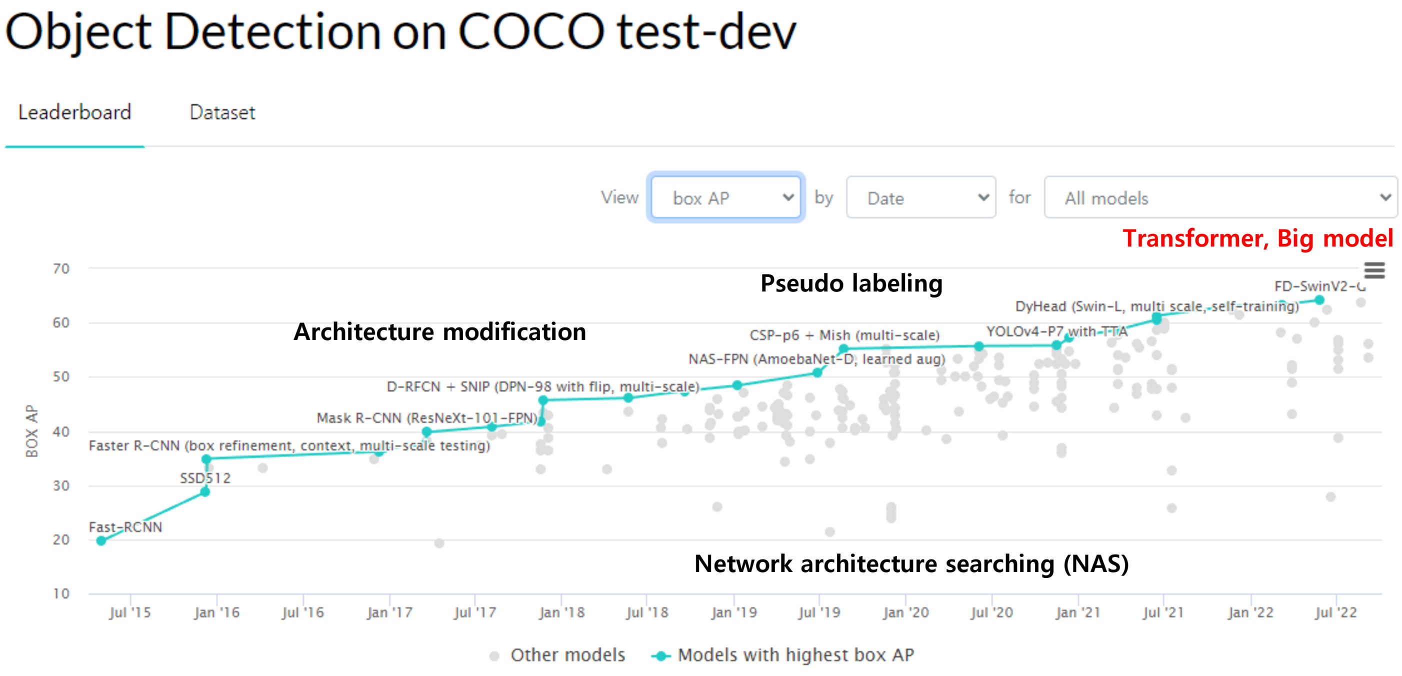
https://github.com/connectomicslab/Aneurysm_Detection

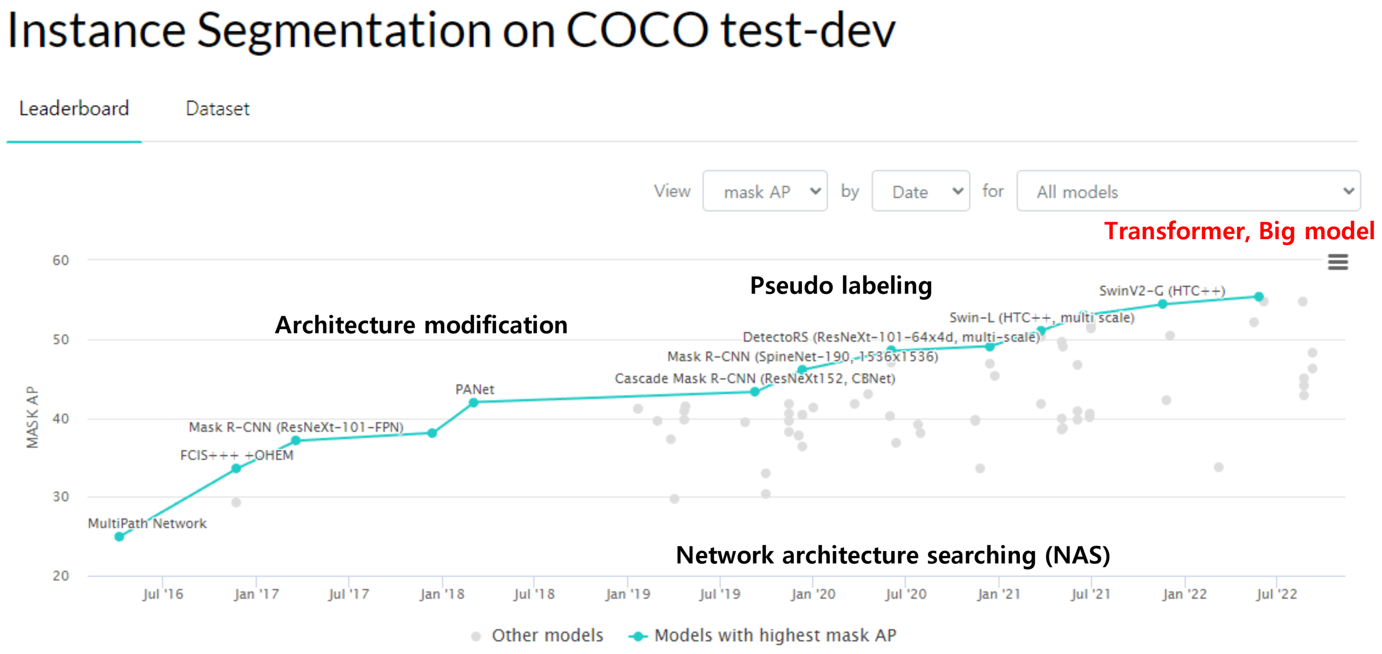
Evaluation



Hover-Net: Simultaneous segmentation and classification of nuclei in multi-tissue histology images, Medical image analysis, 2019

Trend





https://paperswithcode.com/sota

Conclusion

- Object detection 모델 성능은 특정 트랜드를 가지며, 지금도 계속 향상되고 있다.
- 모델 개발자 입장 : 모델 한계점을 파악하고, 개선함으로써 성능, 속도를 향상
- 모델 사용자 입장 : 우리 데이터셋 분석 및 현재 테스크에 맞는 모델 선정이 중요!
- Segmentation vs object detection vs instance segmentation

Hands on

- 1. Google colab (https://colab.research.google.com/)
- 2. 노트 열기 – Github – 아래 깃허브 주소 입력

  https://github.com/kevinkwshin/Handson_detection