Object Detection by MXNET

Chuck

Background I

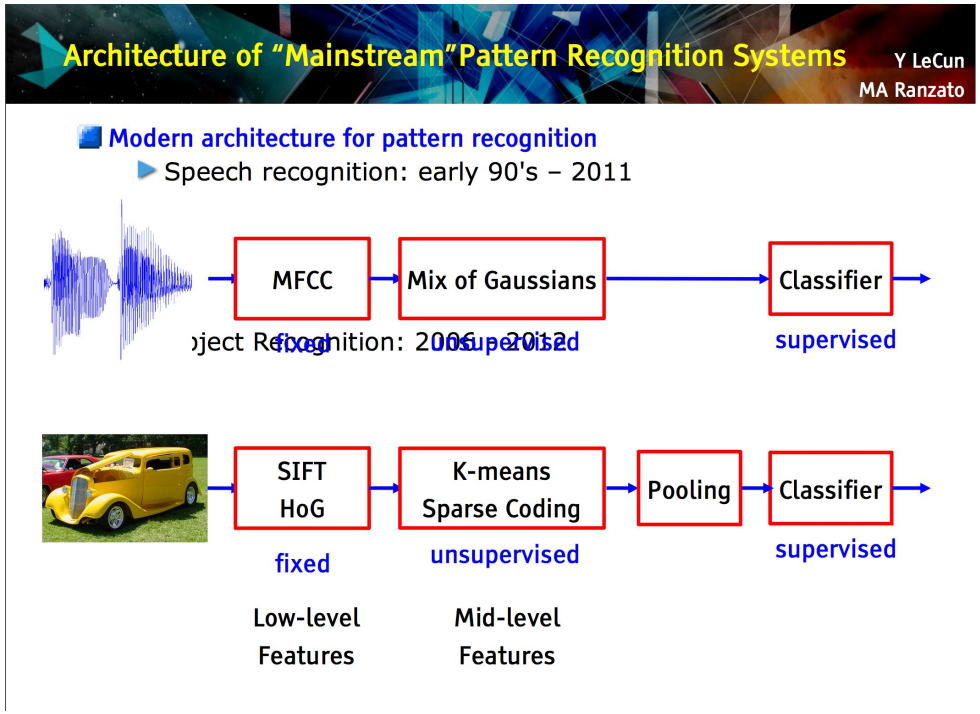
- What is object detection?
  - Object Localization + Object classification
- Why object detection?
  - Self-driving car
  - Security Camera
  - Retailer
  - Robot
  - SkyNet/terminator
  - Many others

Background II

- Why image based Objection Detection (others can be Laser/Rader/LIDAR based)?
  - Cheap camera
- Why can Deep Learning help with this task?
  - Human visual system is fast and accurate to Object Detection task.
  - State-of-art image classification method is deep learning based method.

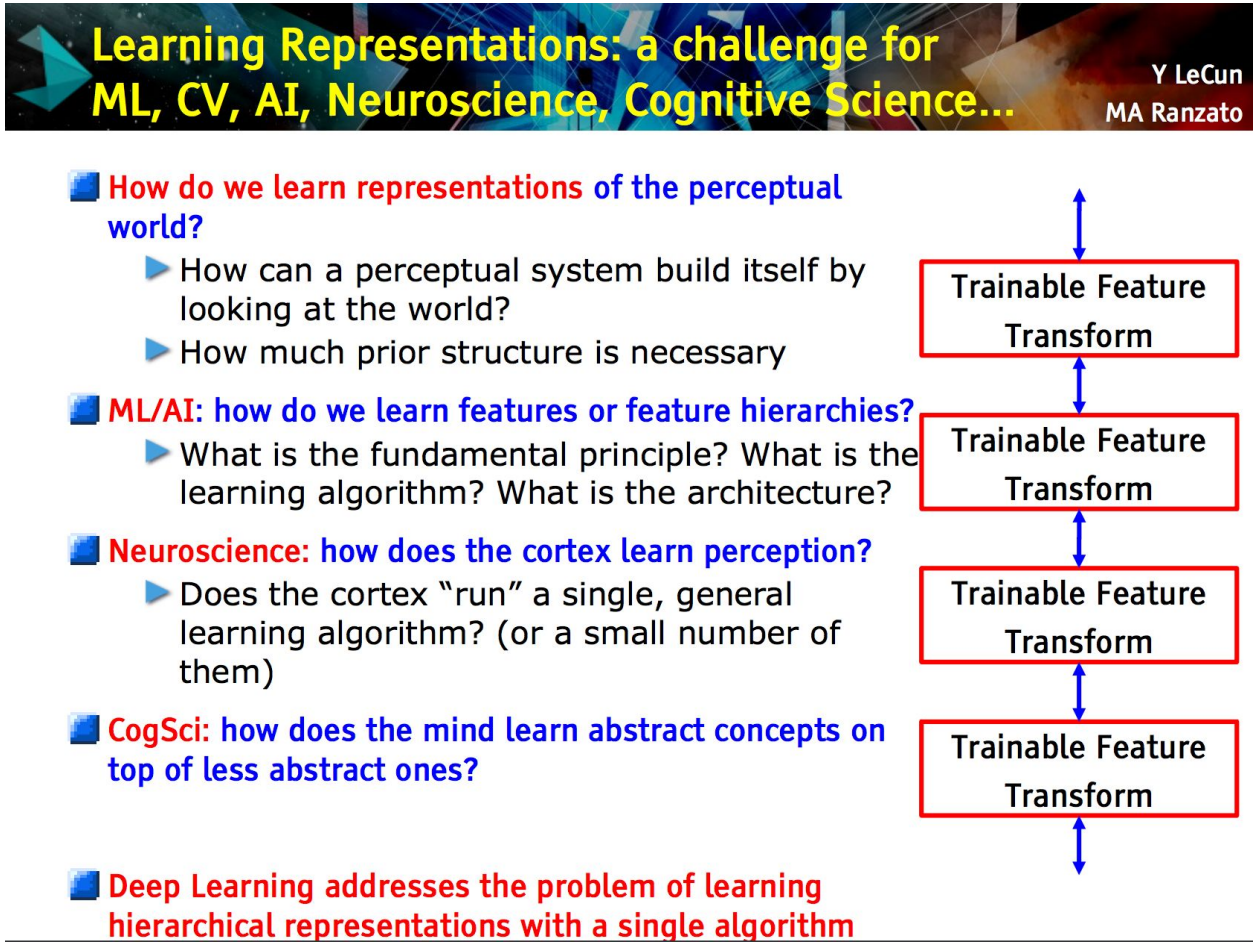
Background III

- Object Detection the traditional way:
  - Get a classifier-> do classification with a sliding window (Ref: Keynote by Yan Lecun in ICML2013)



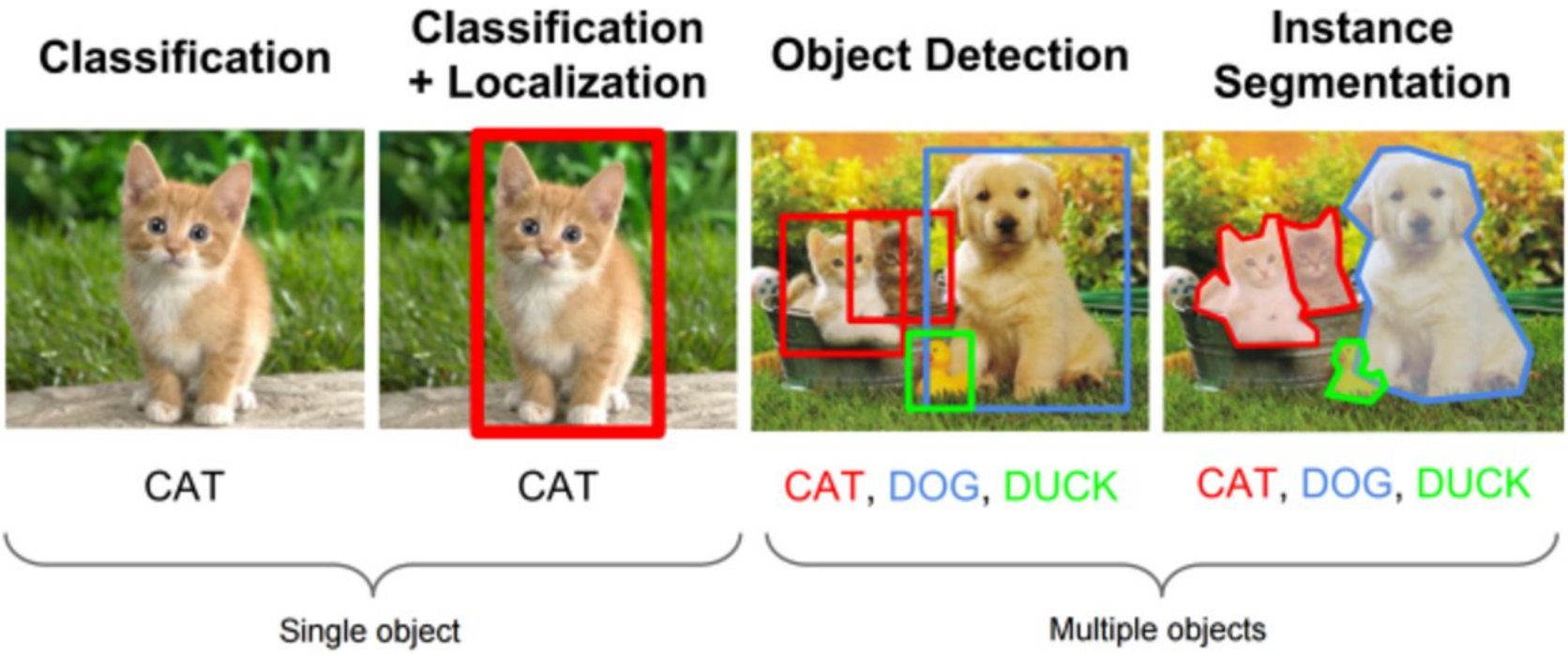
Background IV

- [Keynote by Yan Lecun in ICML2013]



Object Detection with Deep Learning

- What is Object Detection again?



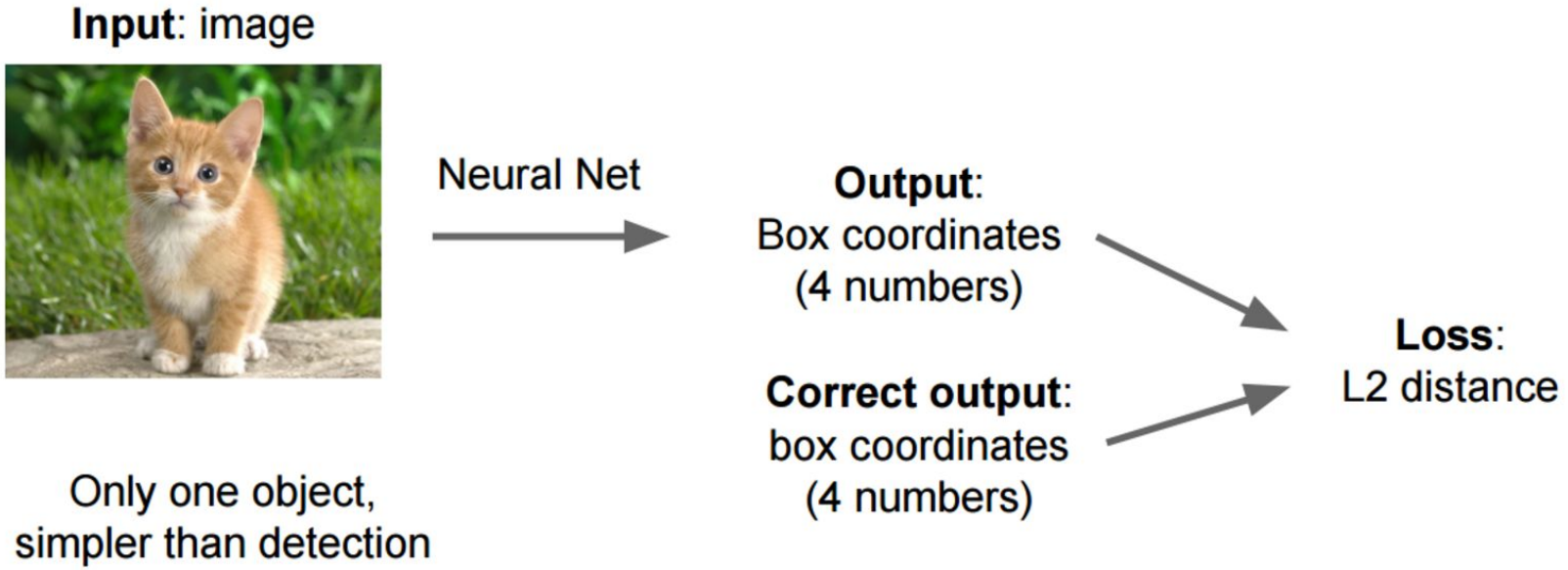
How many different Object Detection Deep Learning methods?

- RCNN
- Fast RCNN
- Faster RCNN
- SSD
- YOLO

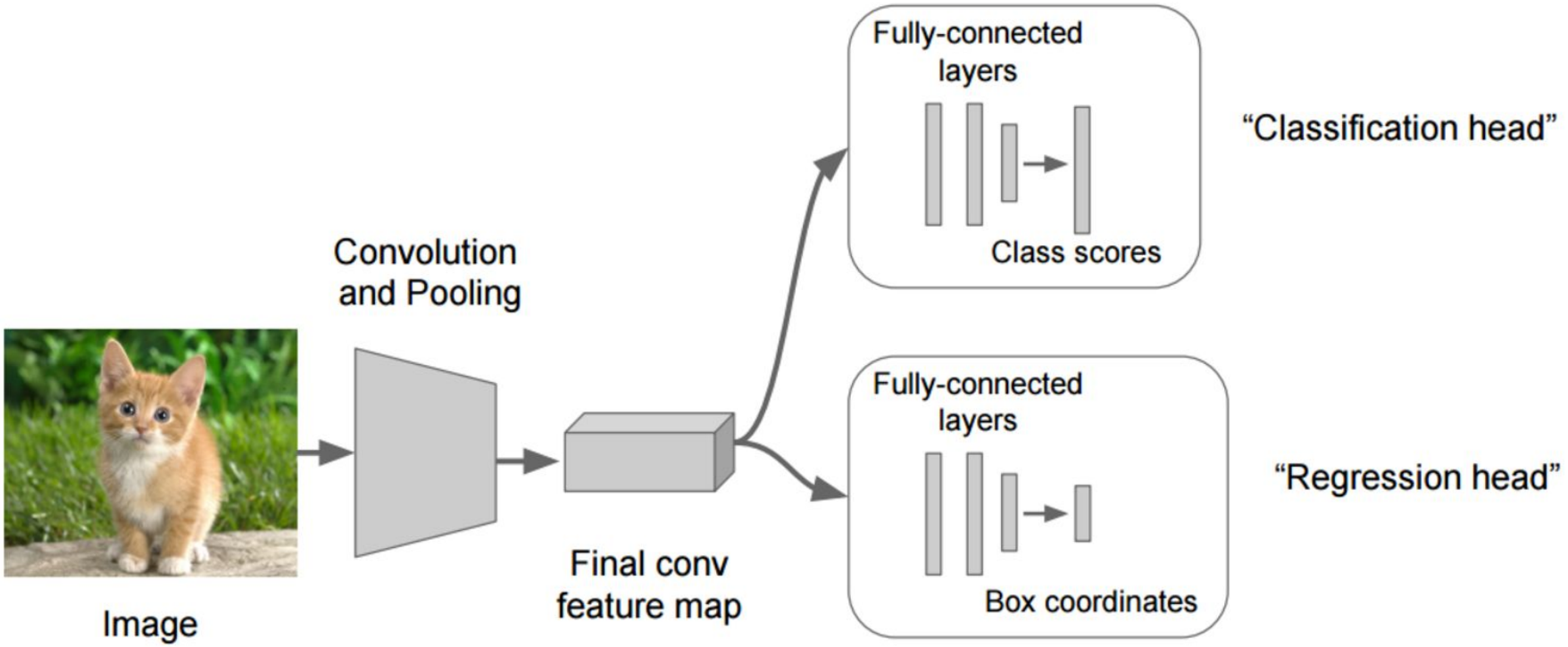
Region Proposal Family

The basic idea

- Classify the object with classification.
- Localize the object with regression.
  - What is regression of object detection?

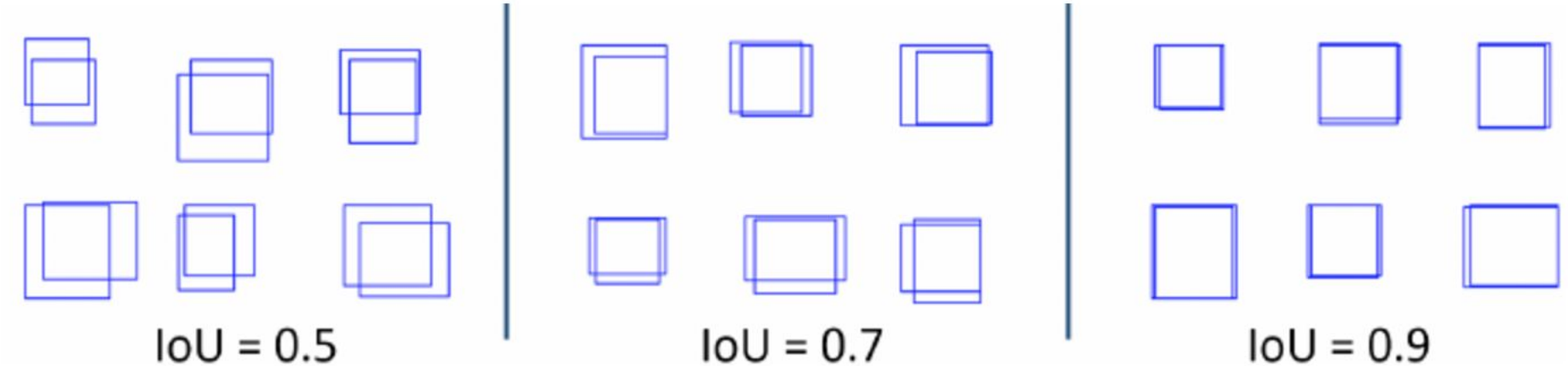


Single Object Localization

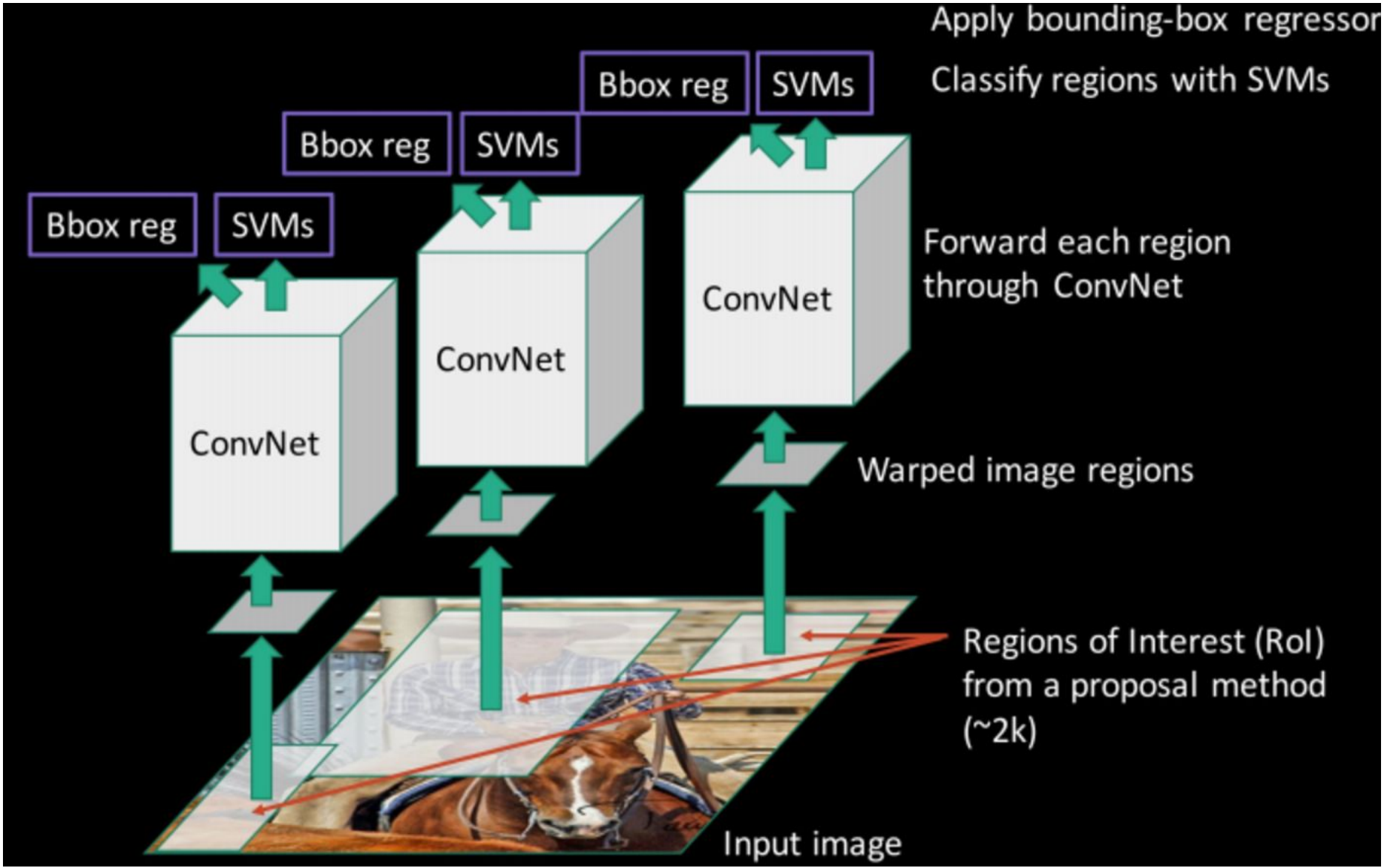


But it can only detect one object each time.

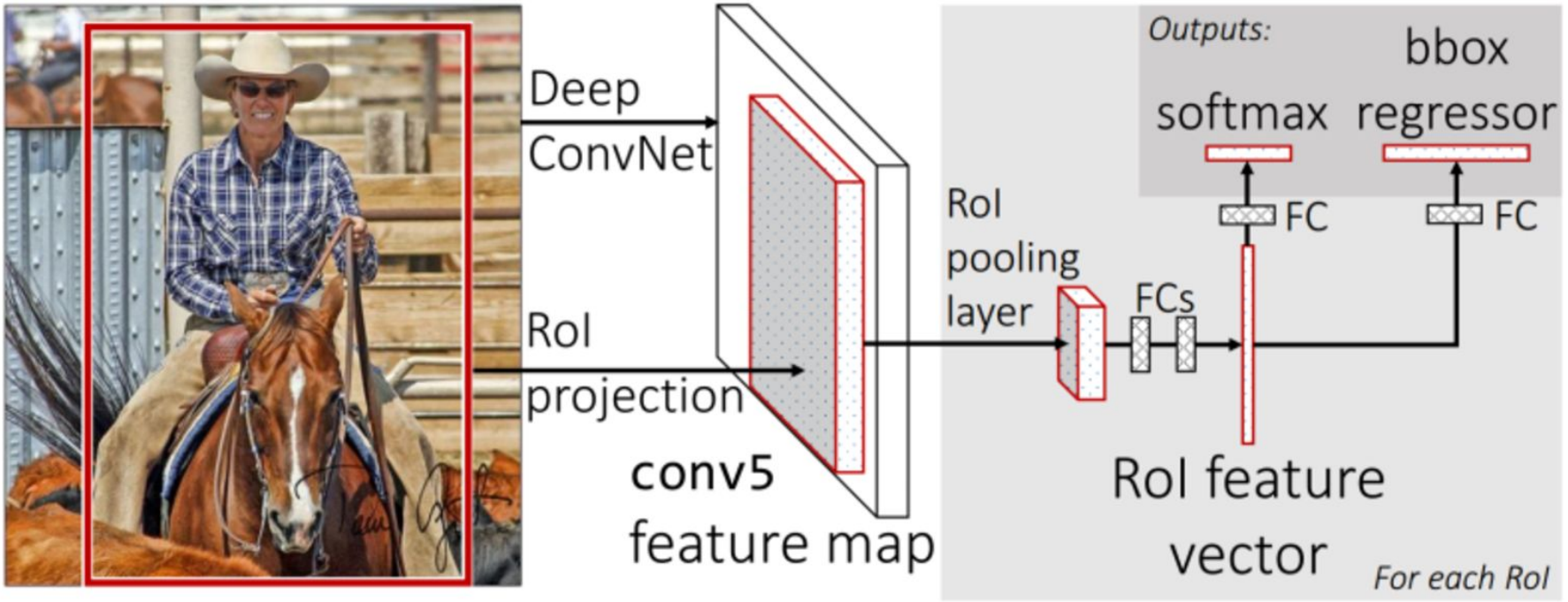
How to measure detection accuracy



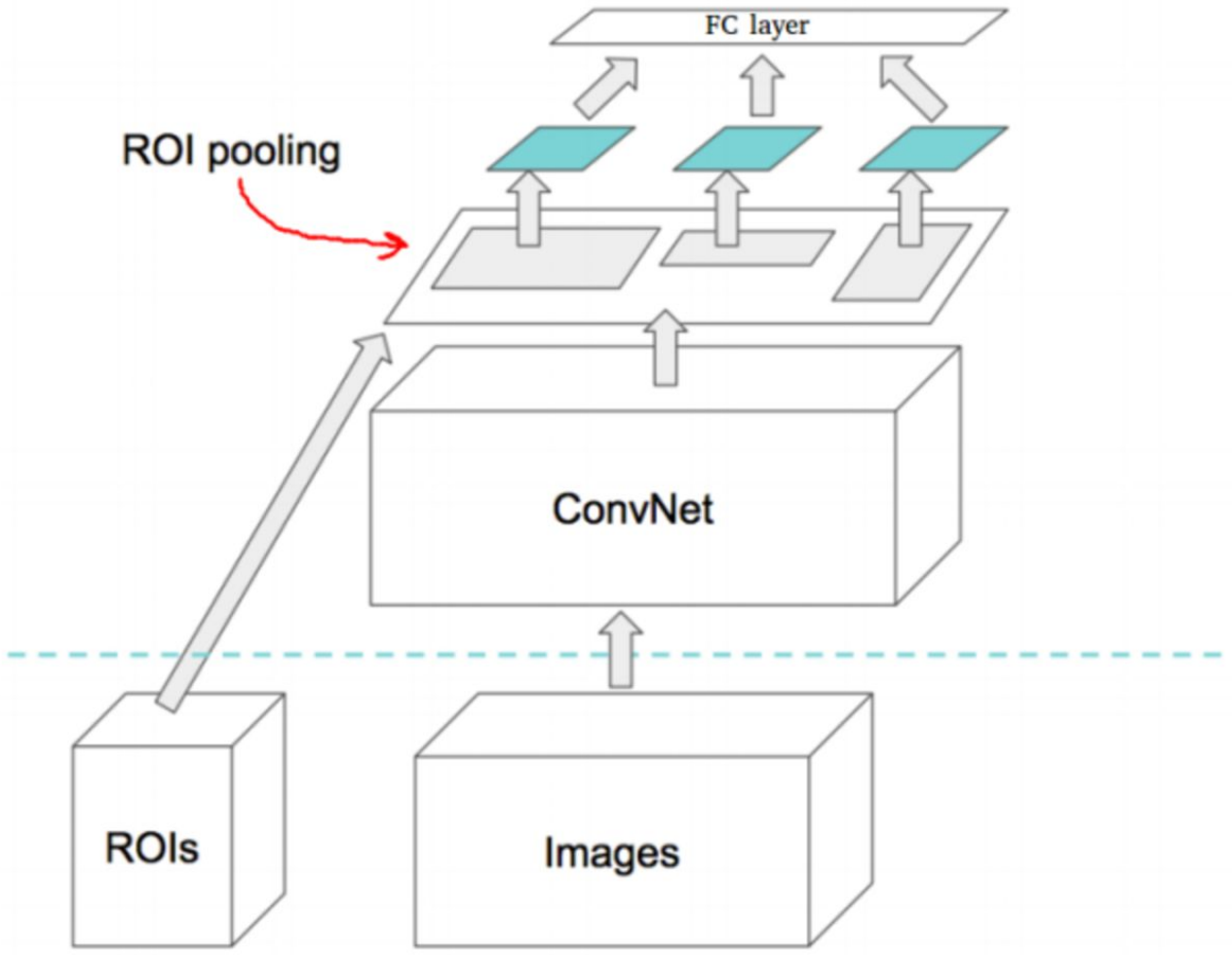
RCNN (really slow)



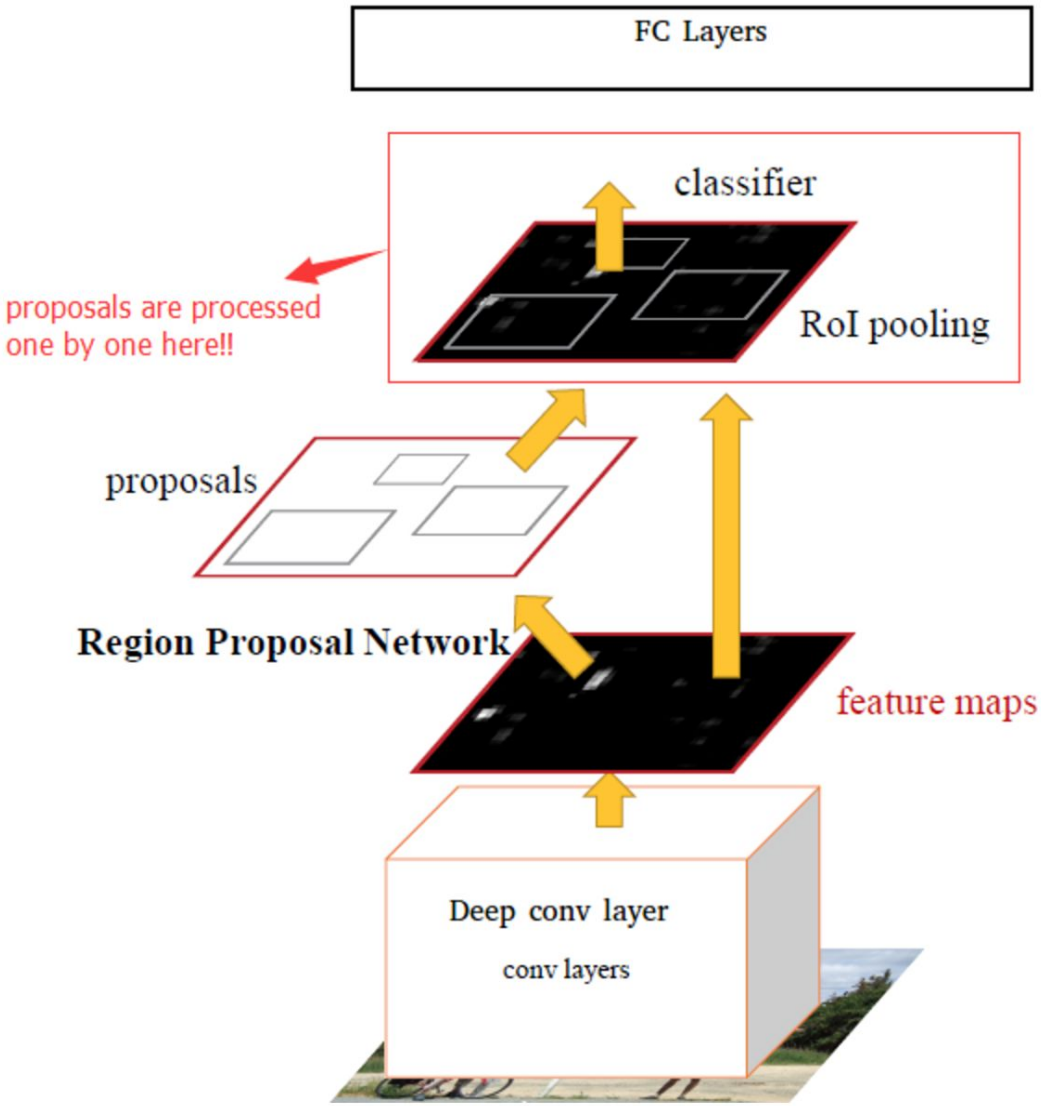
Fast RCNN


Deep
ConvNet
RoI
projection
conv5
feature map
RoI
pooling
layer
FCs
Outputs:
softmax
bbox
regressor
FC
FC
RoI feature
vector
For each RoI

A new layer: ROI layer (hard to compute the gradients)



Faster RCNN(state-of-art)

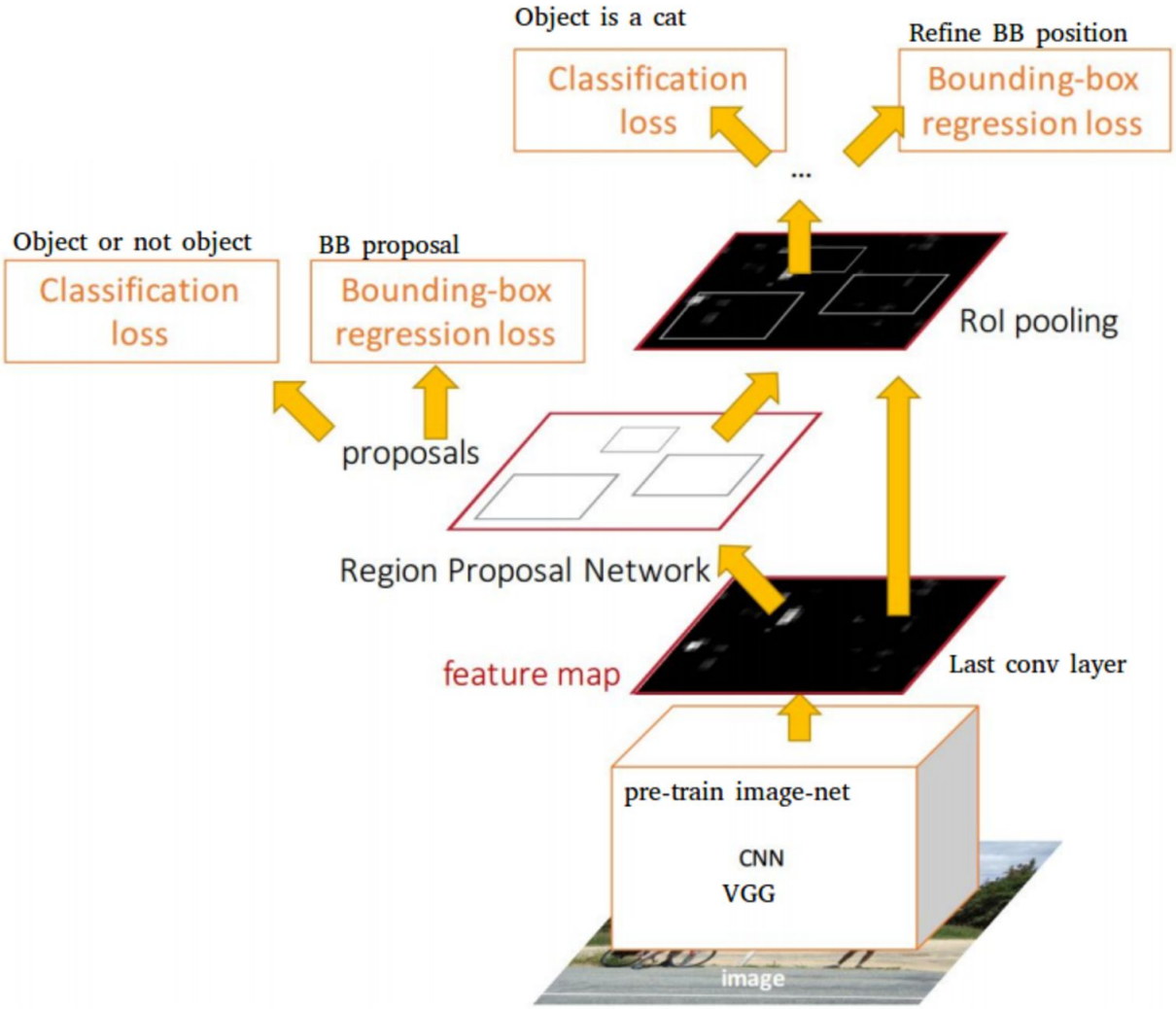


A new layer: Region Proposal Network

- How it works

  1. RPN slides a small window (3x3) on the feature map, that classify what is under the window as object or not object, and also gives some bounding box location.
  2. For every sliding window center it creates fixed k anchor boxes, and classify those boxes as been object or not.

Faster RCNN diagram



Speed Comparison

Faster RCNN results

The best result now is Faster RCNN with a resnet 101 layer.

	R-CNN	Fast R-CNN	Faster R-CNN
Test time per image (with proposals)	50 seconds	2 seconds	**0.2 seconds**
(Speedup)	1x	25x	**250x**
mAP (VOC 2007)	66.0	**66.9**	**66.9**

Pros & Cons

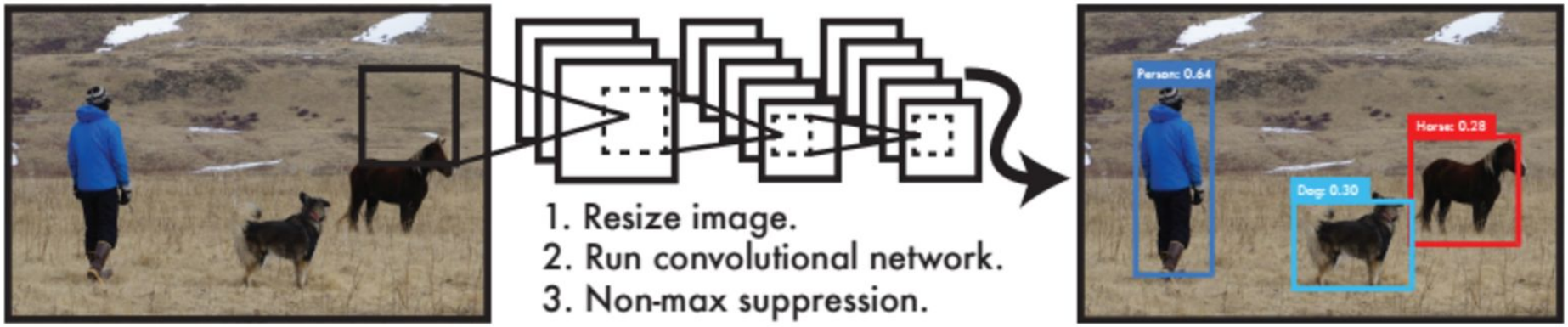
- Pros:
  - Accurate (state-of-art)
- Cons:
  - Slow, need much memory and computation expensive (not friendly to embed device)

So, we want to be faster but ‘good’ accuracy in our final project.

Everything in one network

- The family:
  - SSD: Uses different activation maps (multiple-scales) for prediction of classes and bounding boxes
  - YOLO (v1, v2): Uses a single activation map for prediction of classes and bounding boxes
  - R-FCN(Region based Fully-Convolution Neural Networks): Like Faster RCNN (400ms), but faster (170ms) due to less computation per box also it's Fully Convolutional (No FC layer)

YOLO (v1)


1. Resize image.
2. Run convolutional network.
3. Non-max suppression.
Person: 0.64
Horse: 0.28
Dog: 0.30

Understand YOLO

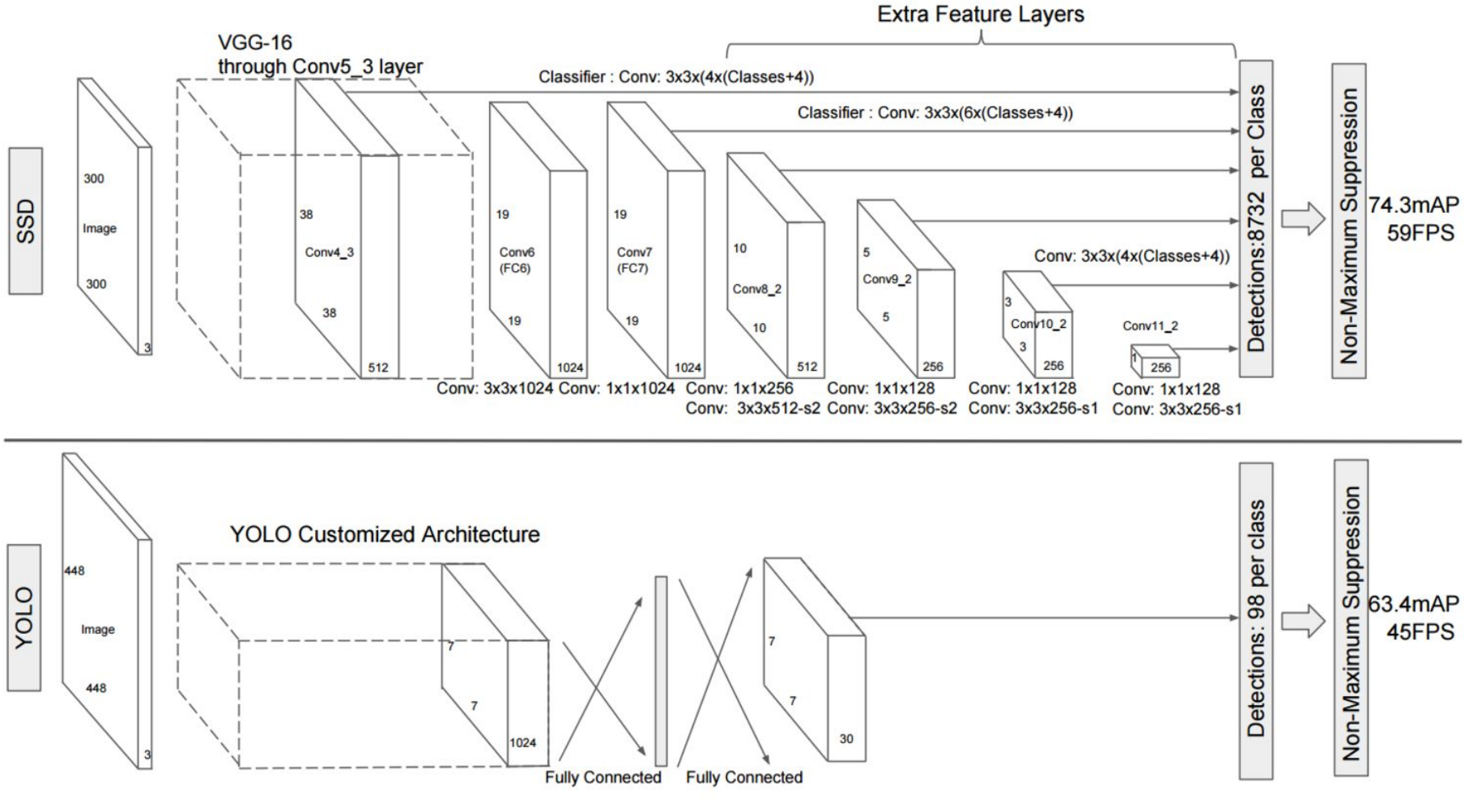
- Paper: https://pjreddie.com/media/files/papers/yolo.pdf
- A good blog to explain YOLO:

  https://medium.com/diaryofawannapreneur/yolo-you-only-look-once-for-object-detection-explained-6f80ea7aaa1e

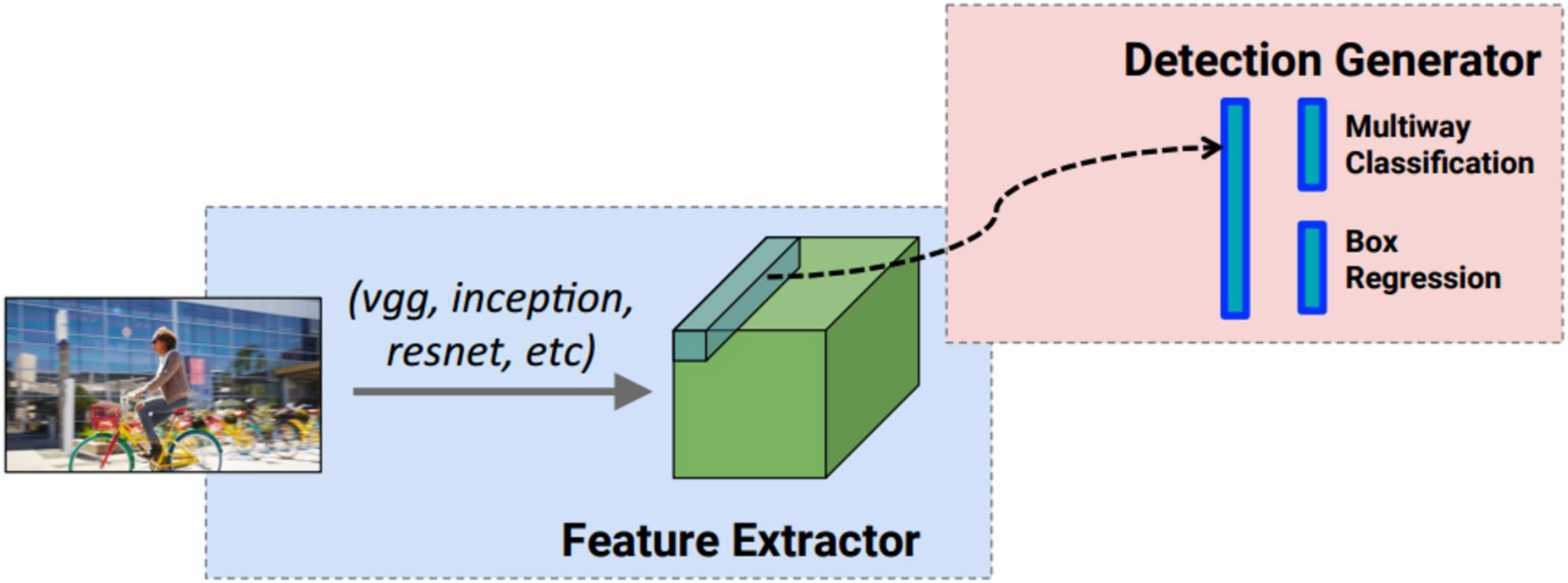
More

YOLO V2: https://pjreddie.com/darknet/yolo/

SSD (1)


SSD
300
Image
300
3
VGG-16
through Conv5_3 layer
38
Conv4_3
38
512
19
Conv6
(FC6)
19
1024
19
Conv7
(FC7)
19
1024
10
Conv8_2
10
512
5
Conv9_2
5
256
3
Conv10_2
3
256
Conv11_2
1
256
Extra Feature Layers
Classifier : Conv: 3x3x(4x(Classes+4))
Classifier : Conv: 3x3x(6x(Classes+4))
Conv: 3x3x(4x(Classes+4))
Conv: 3x3x1024
Conv: 1x1x1024
Conv: 1x1x256
Conv: 3x3x512-s2
Conv: 1x1x128
Conv: 3x3x256-s2
Conv: 1x1x128
Conv: 3x3x256-s1
Conv: 1x1x128
Conv: 3x3x256-s1
Detections:8732 per Class
Non-Maximum Suppression
74.3mAP
59FPS
YOLO
448
Image
448
3
YOLO Customized Architecture
7
7
1024
Fully Connected
Fully Connected
7
7
30
Detections: 98 per class
Non-Maximum Suppression
63.4mAP
45FPS

SSD (2)


Detection Generator
Multiway Classification
Box Regression
(vgg, inception, resnet, etc)
Feature Extractor

MXNet Project

- Cat Detection
- Goal: Build a neural network according to YOLO paper.
- More details next Tuesday.