University of Stavanger

Image Classification and Object detection

Øyvind Meinich-Bache
Associate Professor II

Classification vs detection

Image classification

«car»

Classification with localization

«car»

Object detection

Multiple objects

1 object

Image Classification

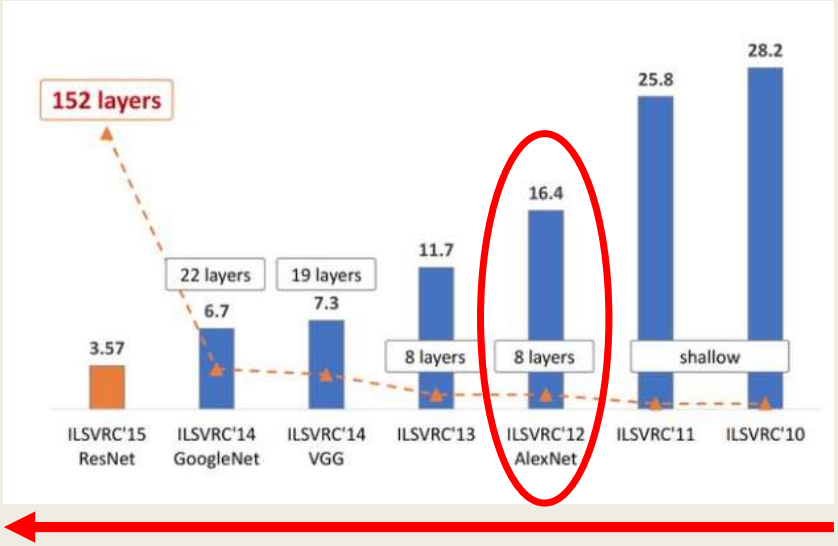
All images in the dataset are only assigned to **one** class – «car»

ØMB1

Image classification history

ImageNet Large Scale Visual Recognition Challenge

2012 winner- AlexNet
- First «deep» model
- Utilized GPU

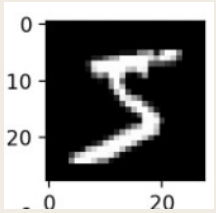


Today: State-of-the-art - Deep convolutional Neural Networks

Slide 4

ØMB1 Øyvind Meinich-Bache; 30.08.2021

Image Classification with MLP



MNIST dataset: http://yann.lecun.com/exdb/mnist/

Image Classification with CNN

CNNs:

- Good at data with spatial information – filters learn spatial features

- Parameter/weight sharing
  -> efficient

Image Classification with CNN

Low-level features → Mid-level features → High-level features → Trainable classifier → Predictions

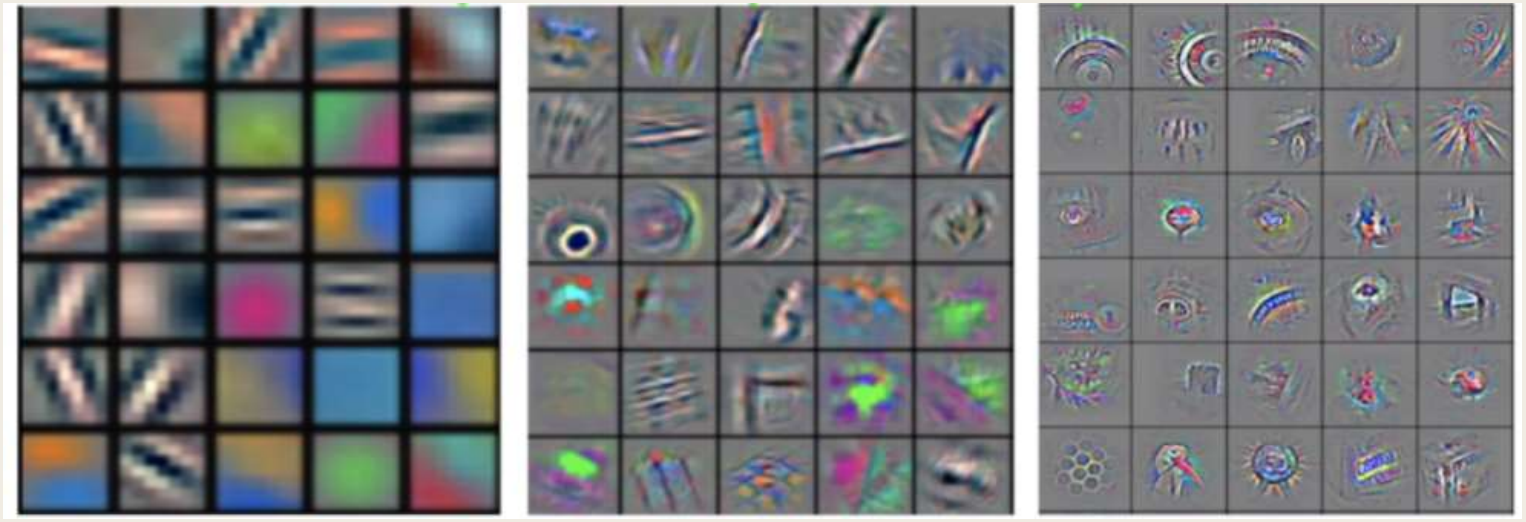
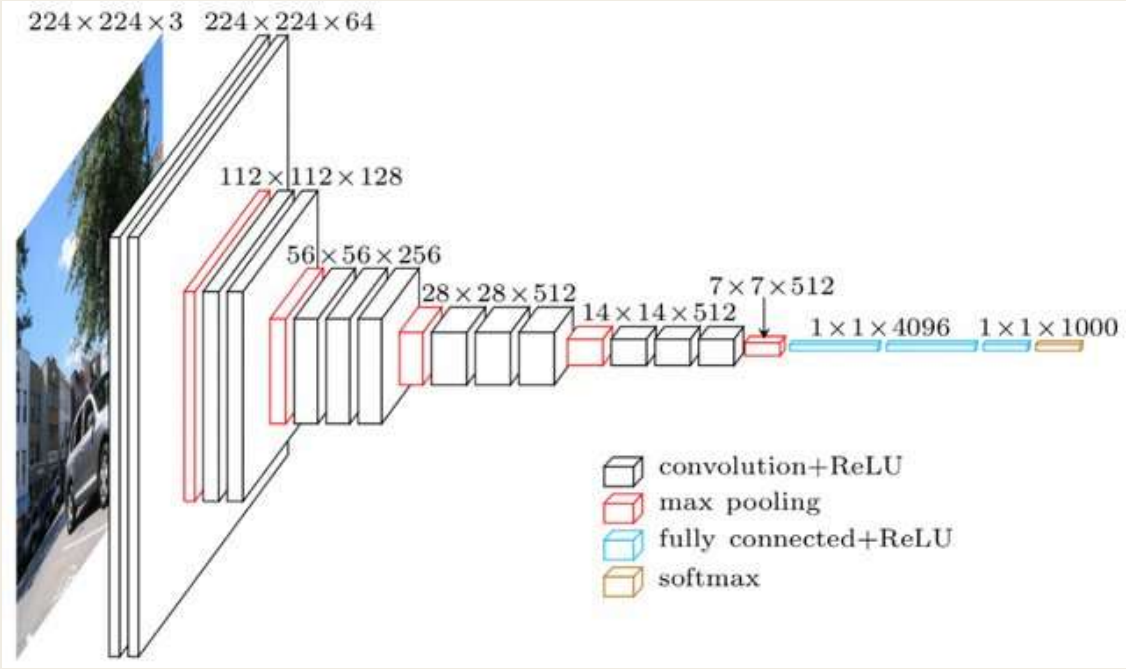


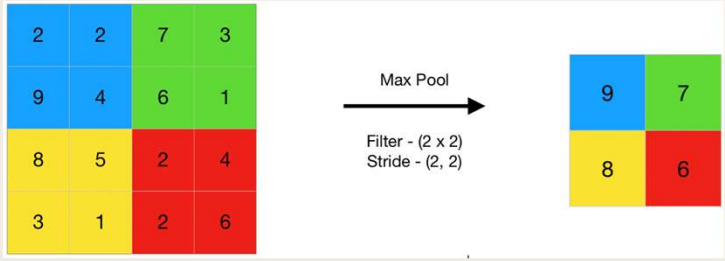
Image Classification – network structures

VGG-16 Network



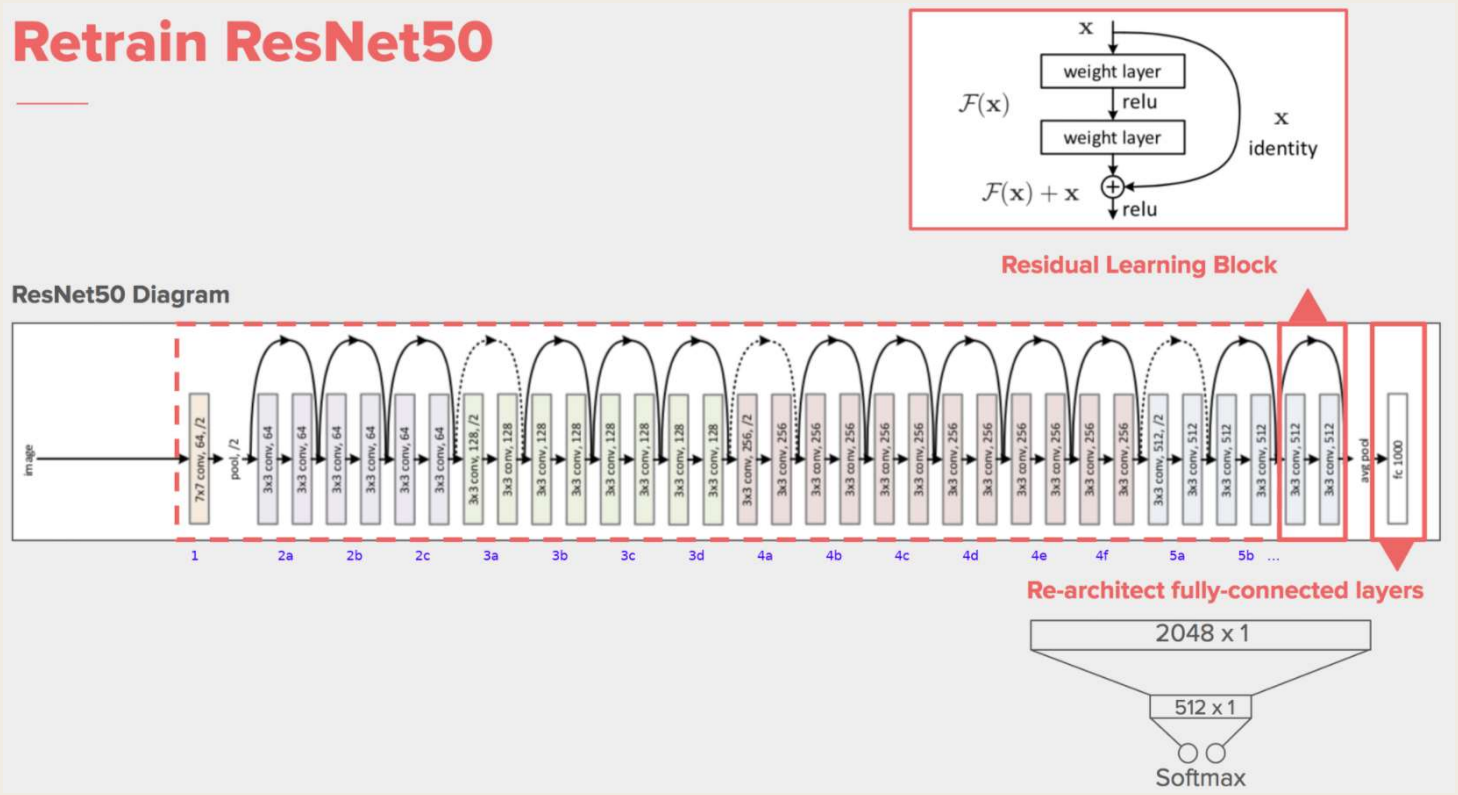
Uses smaller receptive fields in early layers, 3x3 with stride of 1 compared to AlexNet which use 11x11 and stride of 4.

Winner of ImageNet 2014.



Source: https://towardsdatascience.com/step-by-step-vgg16-implementation-in-keras-for-beginners-a833c686ae6c

Image Classification – network structures



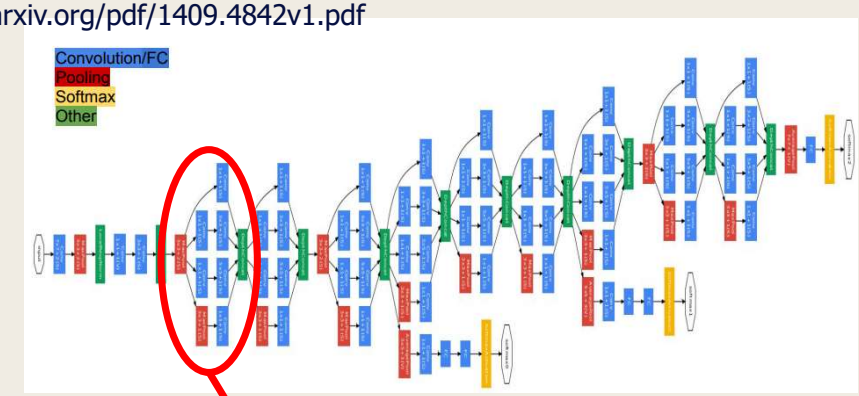
- ResNetet50 have 50 layers
- Often used as starting point for transfer learning for image classification problems
- Also a 152 layer version of ResNet (very deep CNN) - ImageNet winner of 2015
- Handle vanishing gradient problem during training using the «skip connection architecture»

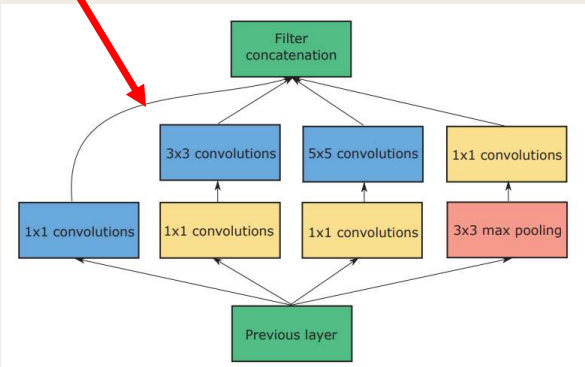
Source: https://i.stack.imgur.com/gI4zT.png

Image Classification – network structures

Inception Network

Source: https://arxiv.org/pdf/1409.4842v1.pdf





Too deep CNNs:
- vanishing/exploding gradients
- Computational expensive

Inception:
- Tackle these problems by letting filters of different size operate on the same level
- More «wider» and less «deeper»

Image Classification – network structures

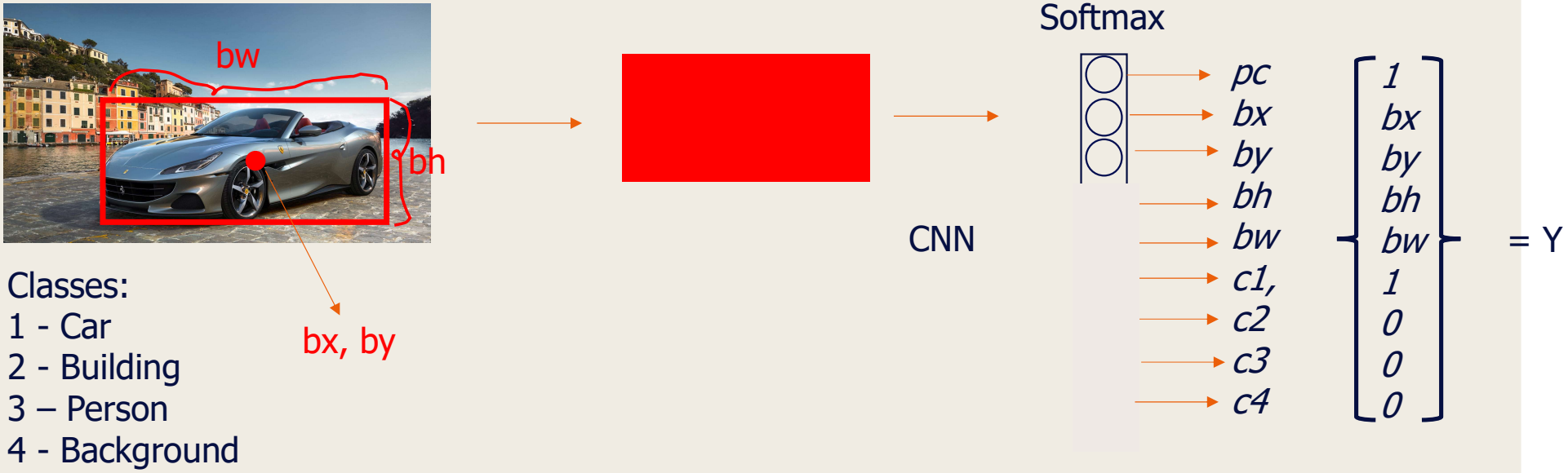
VGG: https://arxiv.org/pdf/1409.1556.pdf
Resnet: https://arxiv.org/pdf/1512.03385.pdf
Inception: https://arxiv.org/pdf/1409.4842.pdf

Optional material:

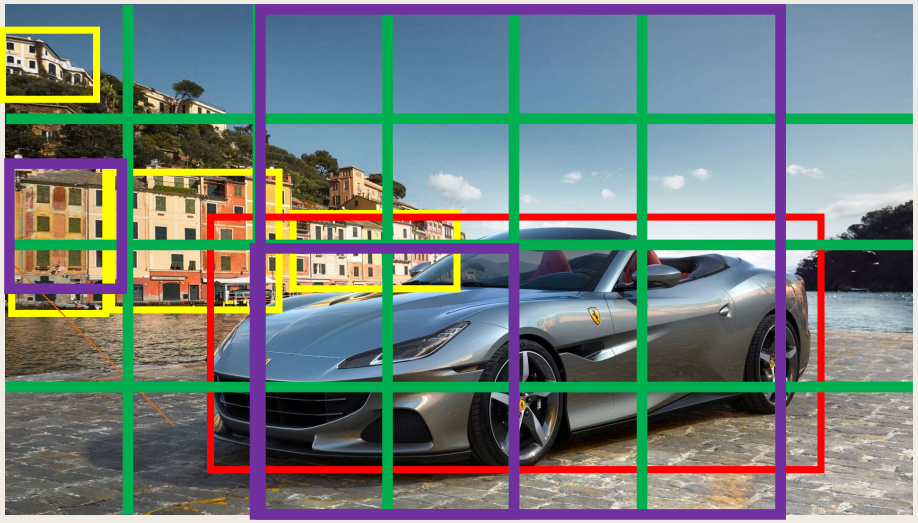
ViT Networks:
Scaling vision transformers: https://arxiv.org/pdf/2106.04560v1.pdf

Image Classification on ImageNet:
https://paperswithcode.com/sota/image-classification-on-imagenet

Image Classification with localization



Object Detection – sliding window



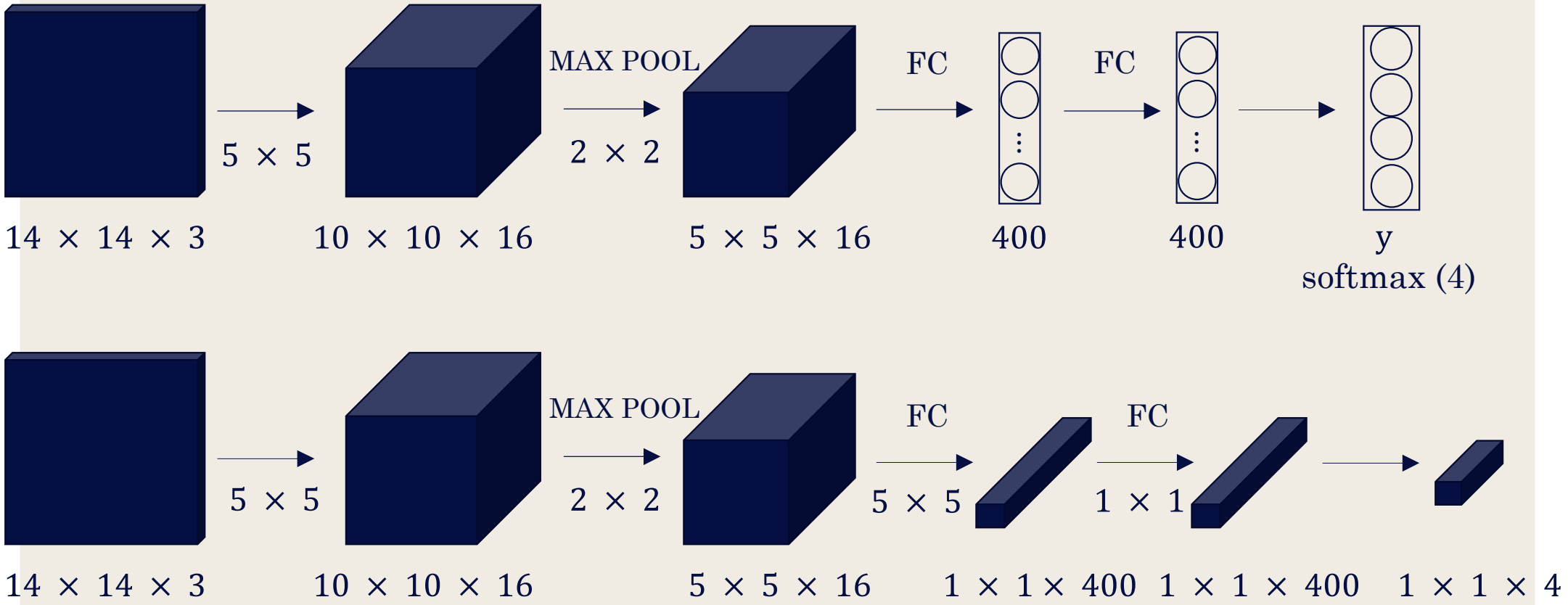
Grid split – image classification in each region

- Objects can cover more than one grid region
  - Need to do a «sliding window» approach

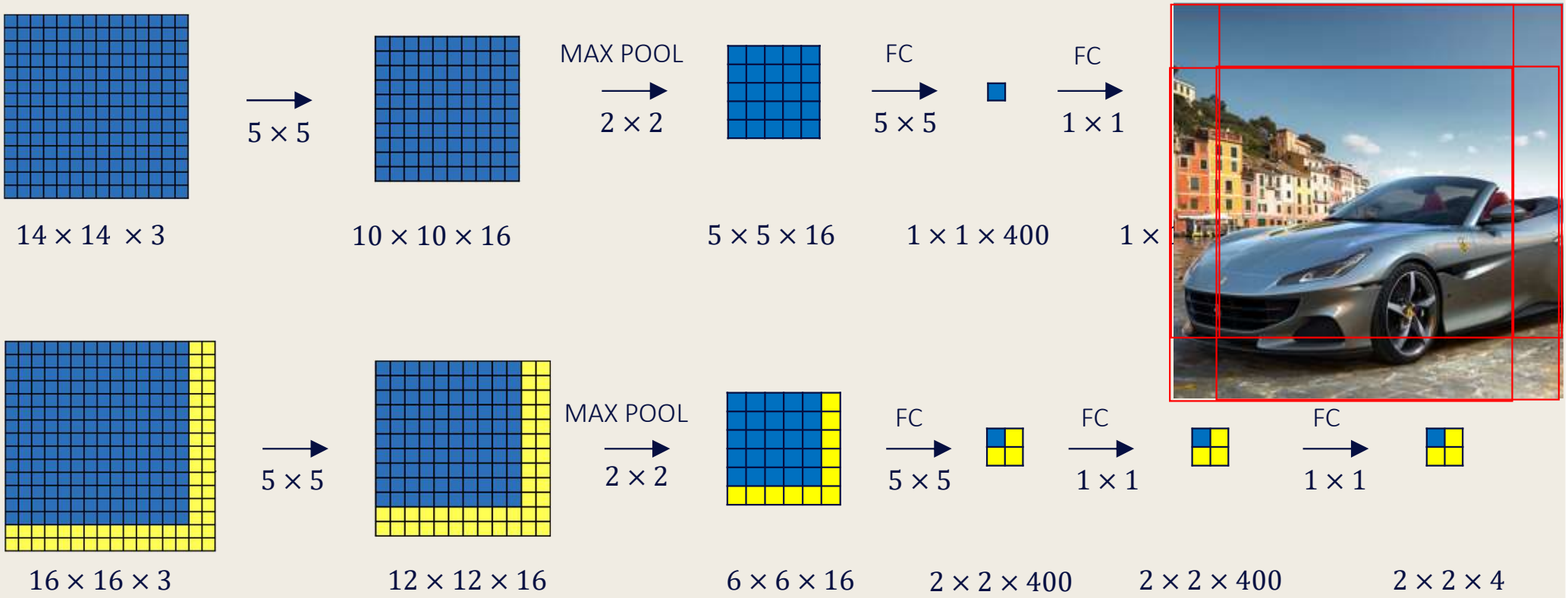
  - Would also need to analyse different sized grid regions

  -> not an efficient approach!

Convolution implementation of sliding windows



Object Detection – Convolution implementation of sliding window



Object Detection

Object Detection

«car»

«car»

Pc = 0.7

Pc = 0.9

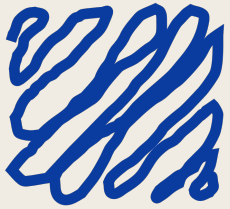
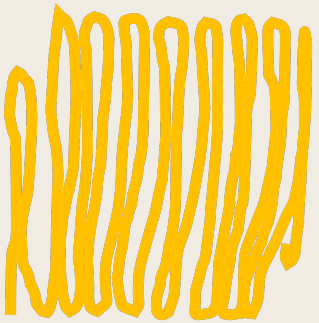
Pc = Prediction probability

If multiple bounding boxes of same object – keep only the one with the highest prediction probability

How?

Object Detection - IuO

Intersection over union (IoU)
- measurement of overlap between two bounding boxes

$$IoU = \frac{\text{Size of } \raisebox{-0.3em}{  }}{\text{Size of } \raisebox{-0.3em}{  }}$$

Object Detection - Non-maximum supression

1: discard all boxes with, pc < Threshold

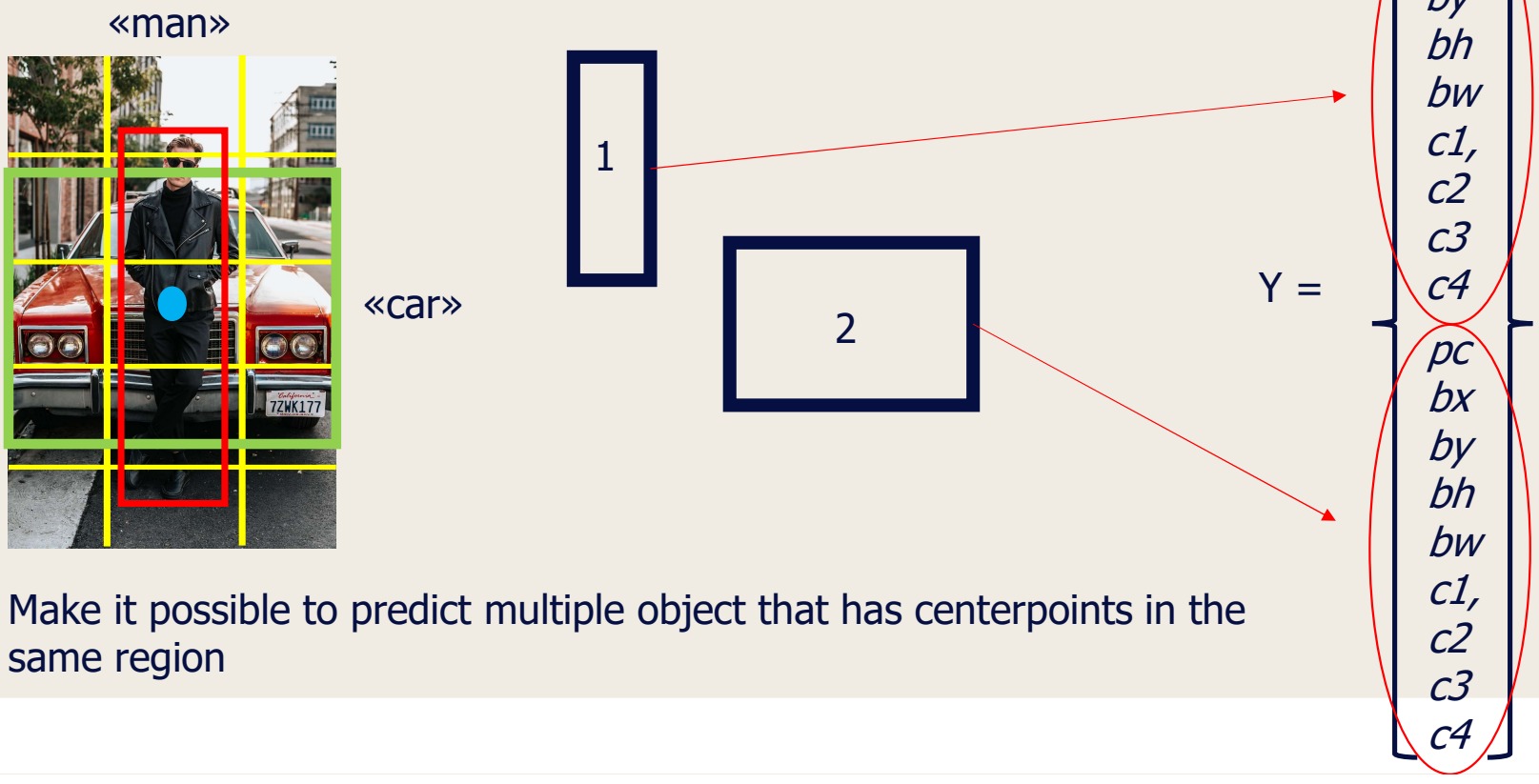
If there are remaining boxes:

2: Pick the box with the largest pc and output that as a prediction

3: discard any remaining boxes with IoU > 0.5

Object Detection – Anchor boxes

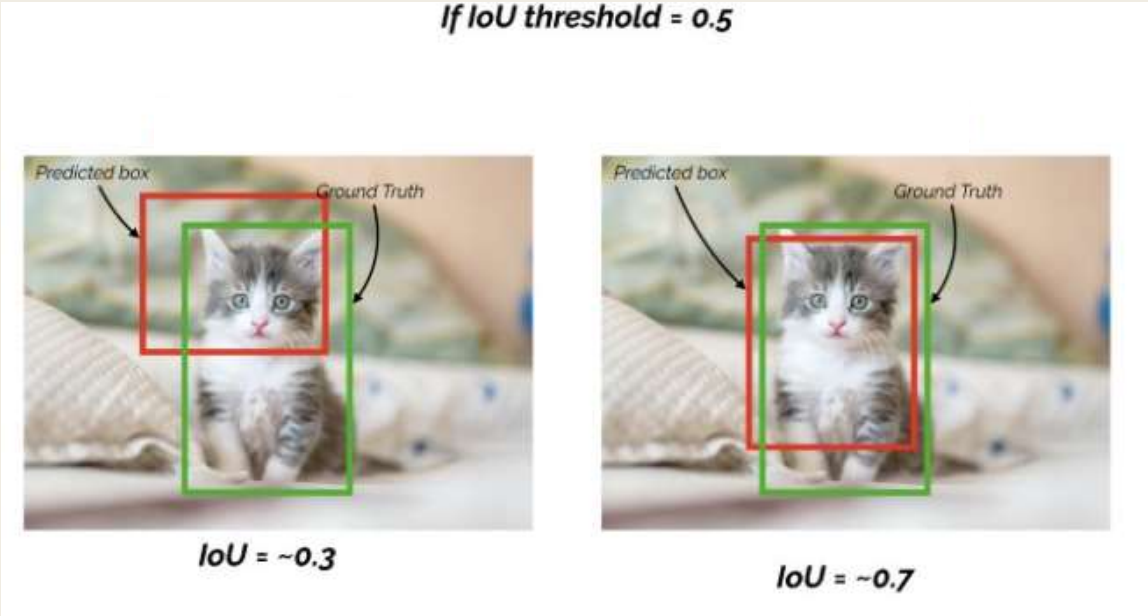
Region vector in output tensor



Make it possible to predict multiple object that has centerpoints in the same region

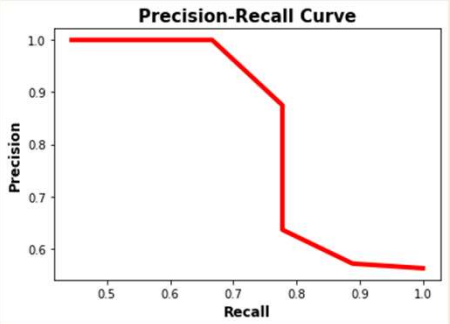
Object Detection – mean Average Precision (mAP)

Object detector performance metric. Estimates an Average Precision for each class and then average over all classes to get the mAP

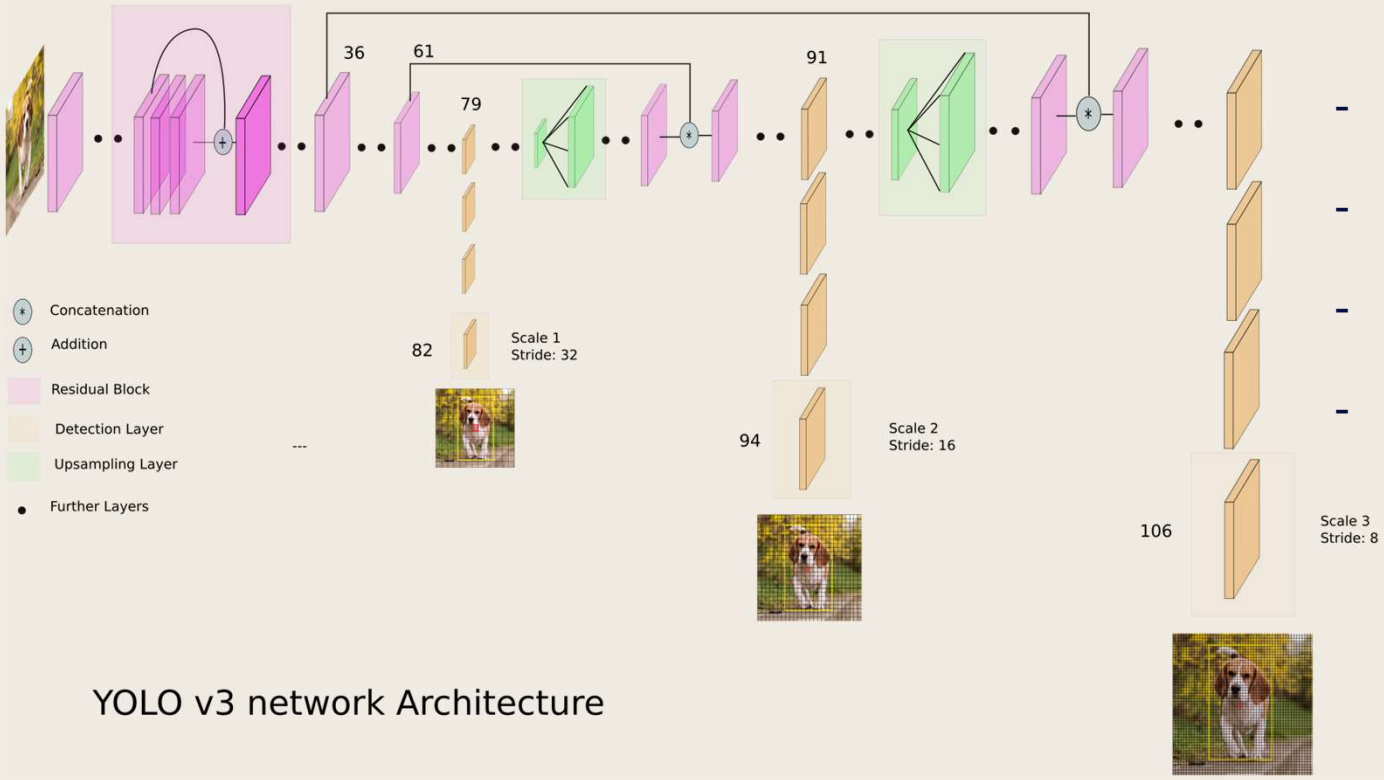


$$Precision = \frac{TP}{TP + FP}$$

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



Object Detection – YOLOv3



YOLO v3 network Architecture

- 9 anchor boxes, 3 for each scale
- Scales - 13x13, 26x26 and 52x52
- -> Output 10 647 predicted boxes
- Non-maximum supression

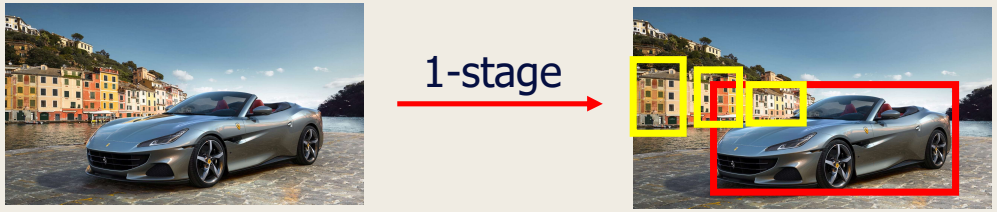
One-stage object detector

Source: https://towardsdatascience.com/yolo-v3-object-detection-53fb7d3bfe6b

Object Detection – one-stage detectors

- YOLO
  - v3 : https://arxiv.org/pdf/1804.02767.pdf
  - v4 : https://arxiv.org/pdf/2004.10934.pdf
  - PP-YOLO : https://arxiv.org/pdf/2007.12099.pdf

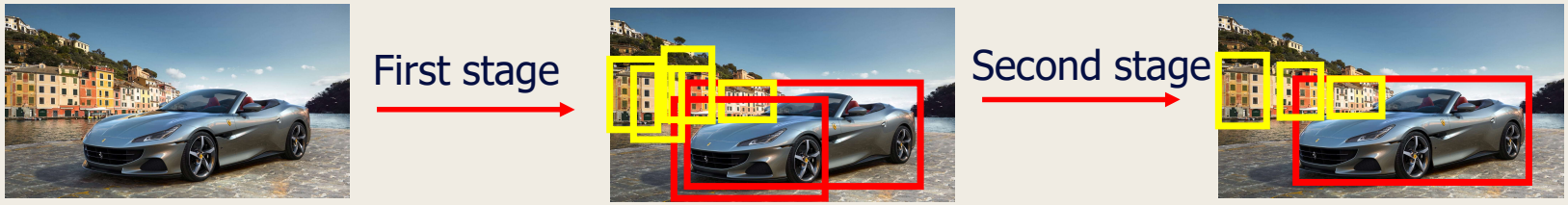
- RetinaNet
  - https://arxiv.org/pdf/1708.02002.pdf



Object Detection – two-stage detectors

Two-stage object detectors:

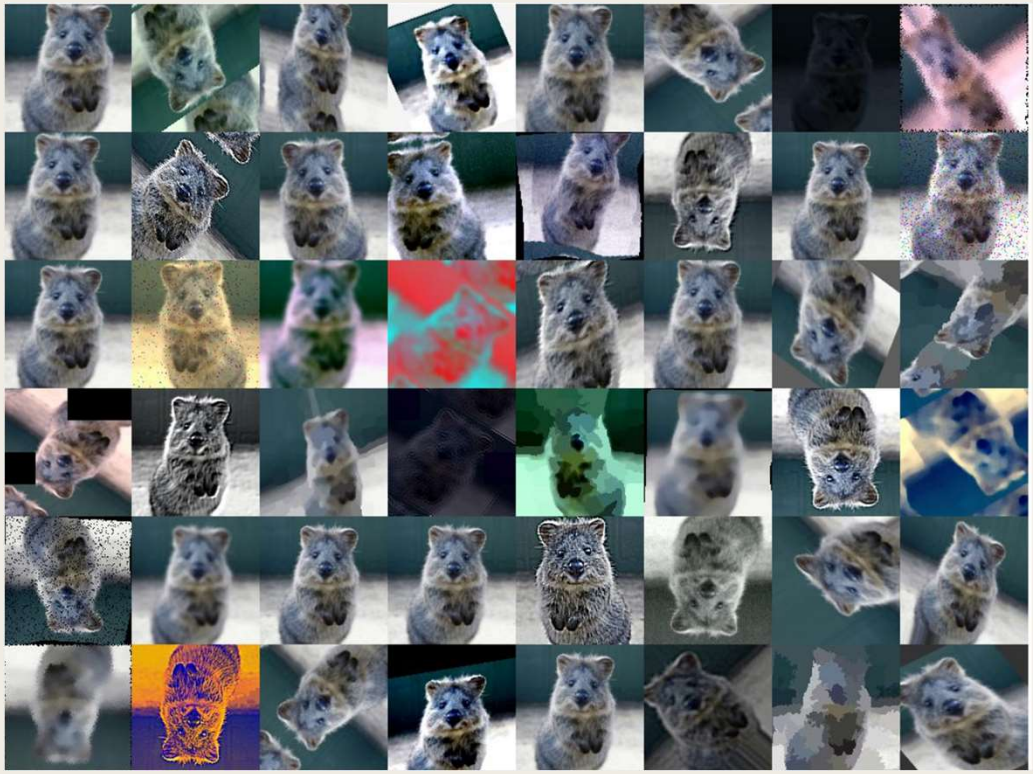
- First step: region proposal,
  Second step: classifcation based on feature extraction from regions
- Normally not end-to end trainable – trained in two steps
- Highest accuracy, but slower. Peformance (fps) drops.



Object Detection - two-stage detectors

- Popular 2-stage object detectors:
  - Region-Based Convolutional Neural Networks (R-CNN) –
    - https://arxiv.org/pdf/1311.2524v5.pdf
  - Faster R-CNN,
    - https://arxiv.org/pdf/1506.01497.pdf
  - Mask R-CNN
    - https://arxiv.org/pdf/1703.06870.pdf

Data Augmentation



Source: https://github.com/aleju/imgaug

- Increase variations and dataset size
- Rotation, cropping, blurring, noise, color adjustment etc

Training, Validation, Test data

Car detector

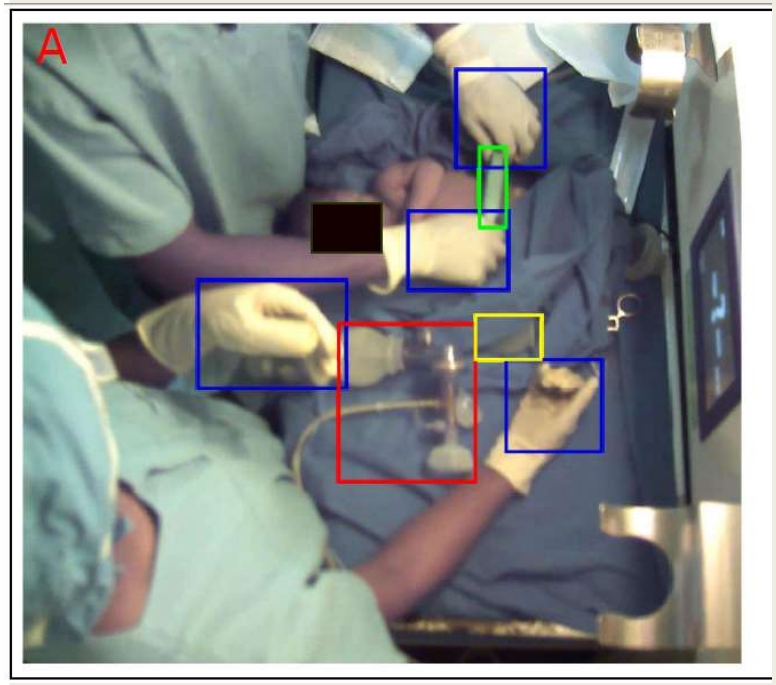
- Need to come from the same distributions!

Training/val data

Model

Test data

Object detection in Newborn Resuscitation Videos



Source: https://pubmed.ncbi.nlm.nih.gov/31247581/