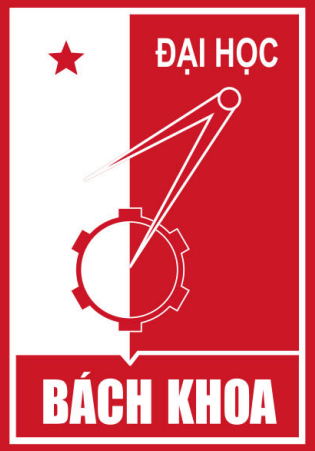


ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

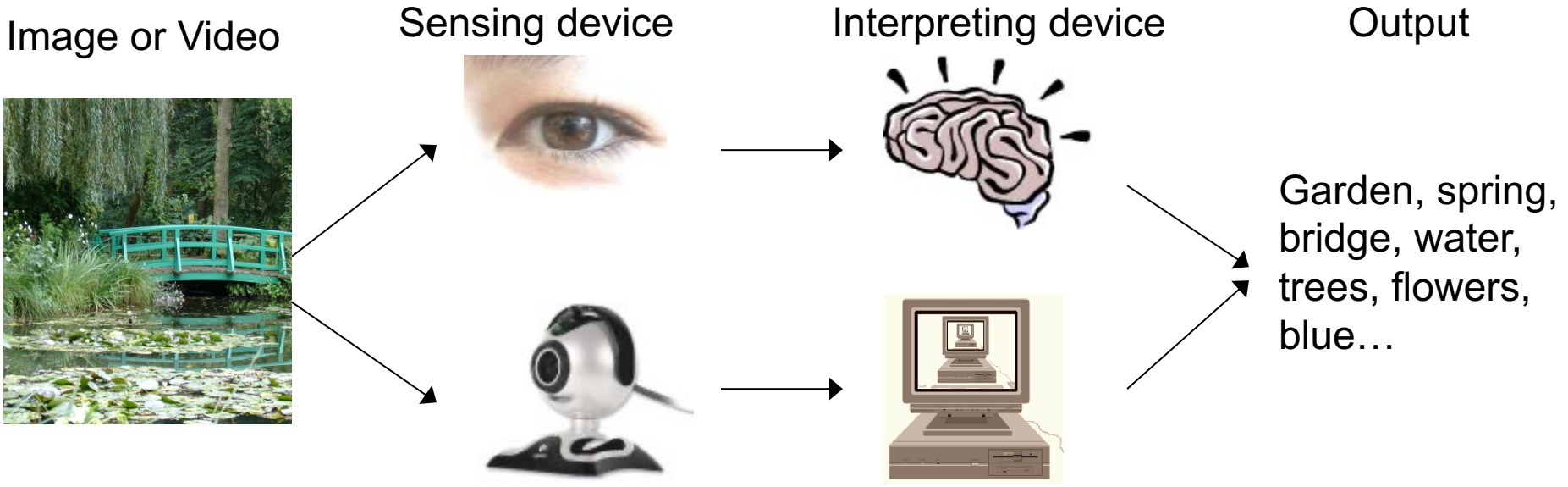
Lesson 6:
Object detection

Outline

1. Overview of computer vision and applications
2. Introduction to object detection
3. Regional proposal networks: R-CNN, Fast R-CNN, Faster R-CNN...
4. None regional proposal networks: SSD, Yolo…

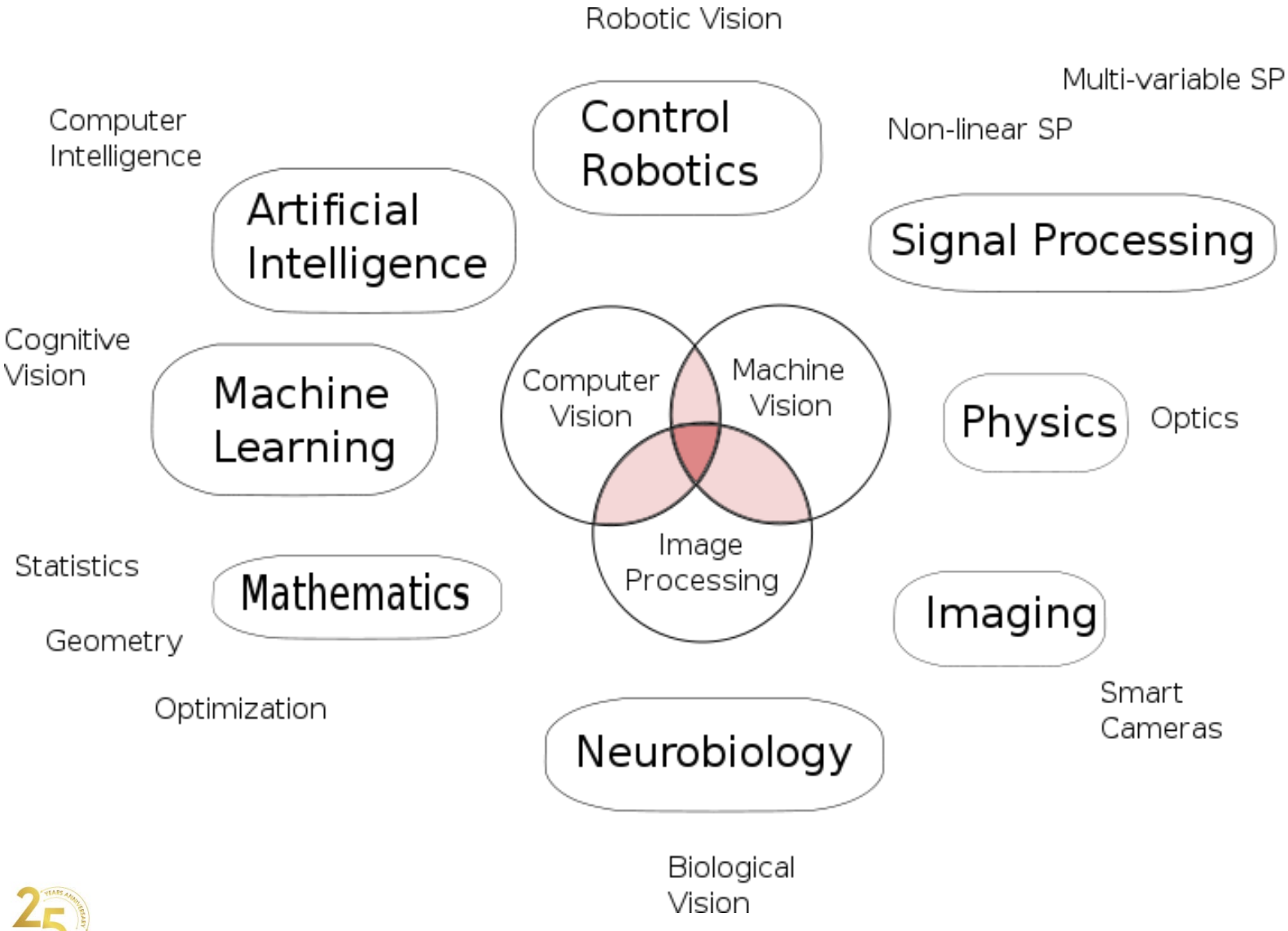
Overview of computer vision and applications

What is computer vision?



- CV is a scientific field that extracts information out of digital images.
- CV Is building algorithms that can understand the content of images and use it for other applications

Overview of relations between computer vision and other fields



Human eyes are very sensitive

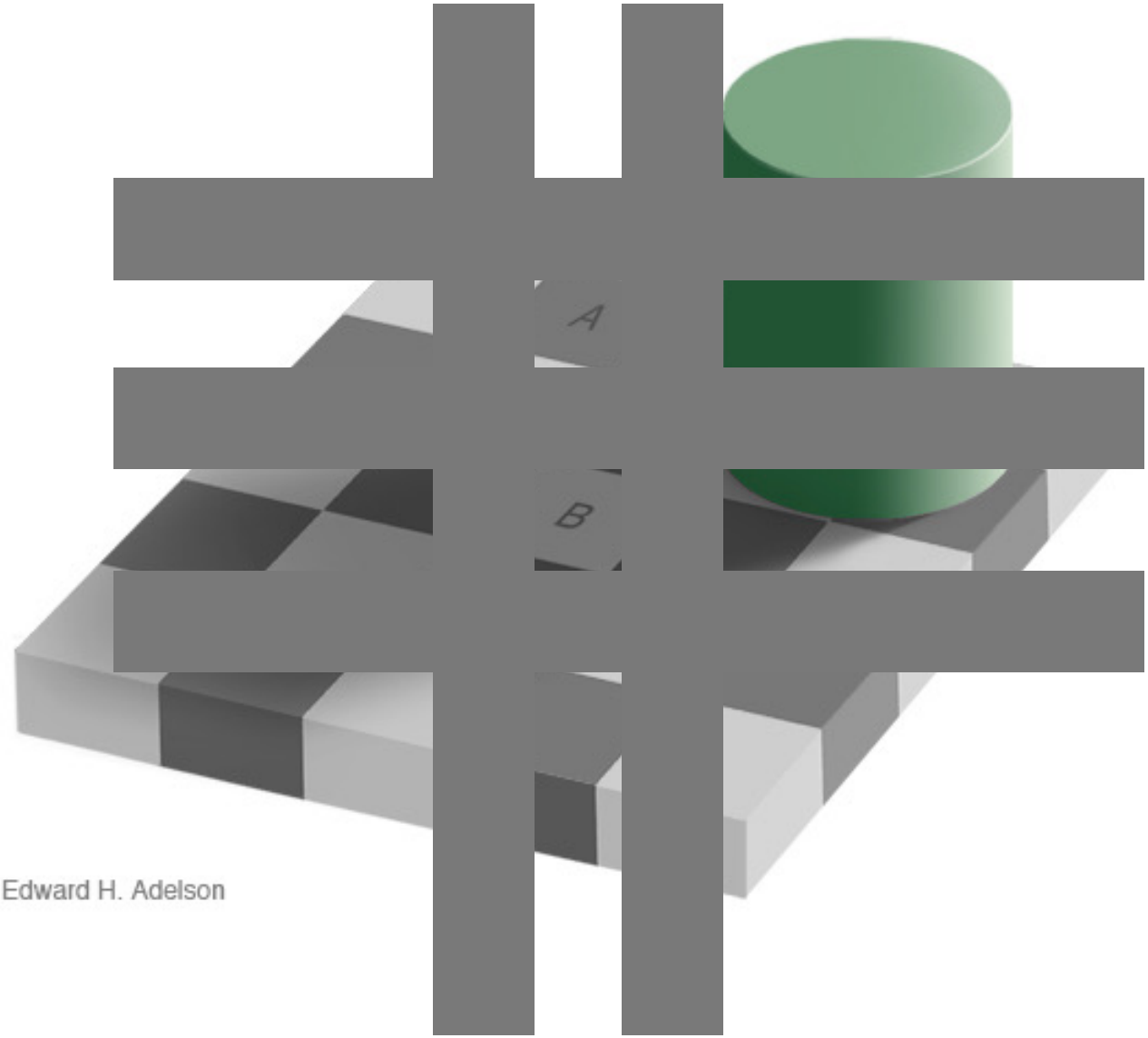
Can u detect a person?
100 ms per frame,
Never seen the picture, do not know the person
Can do effortlessly

Potter, Biederman, etc. 1970s

However, this speed is obtained at the price of some drawbacks

- Can you recognize who is it?

Do you think the colors in boxes A and B are different?



Edward H. Adelson

Objectives of computer vision

- Bridges between numerically represented pixels with semantics

What we see

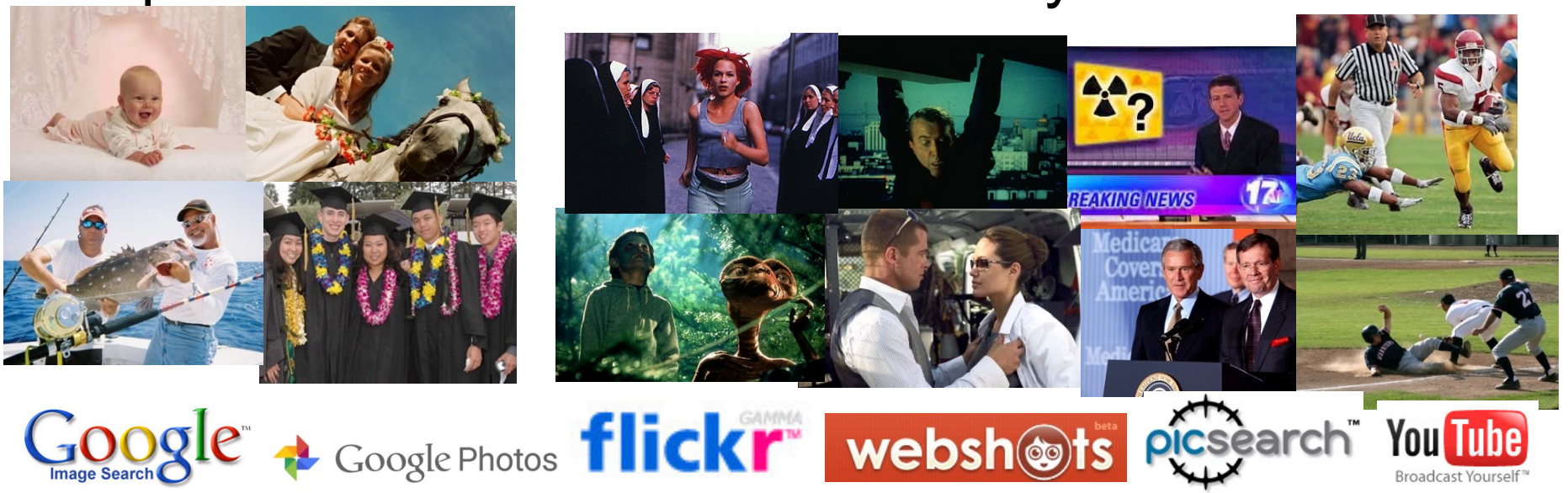
0	3	2	5	4	7	6	9	8
3	0	1	2	3	4	5	6	7
2	1	0	3	2	5	4	7	6
5	2	3	0	1	2	3	4	5
4	3	2	1	0	3	2	5	4
7	4	5	2	3	0	1	2	3
6	5	4	3	2	1	0	3	2
9	6	7	4	5	2	3	0	1
8	7	6	5	4	3	2	1	0

What the computer sees

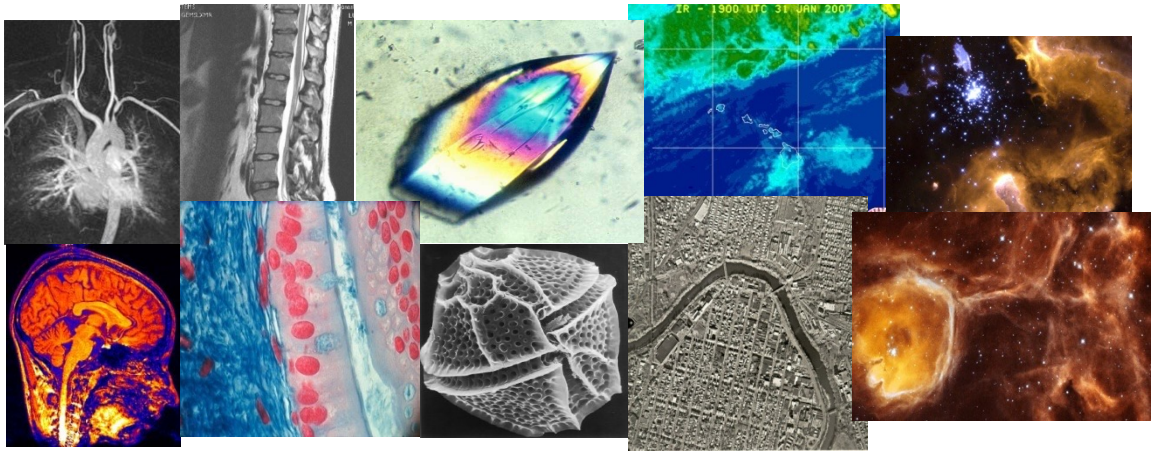
Source: S. Narasimhan

Why learn computer vision?

- Helpful: Photos and videos are everywhere!

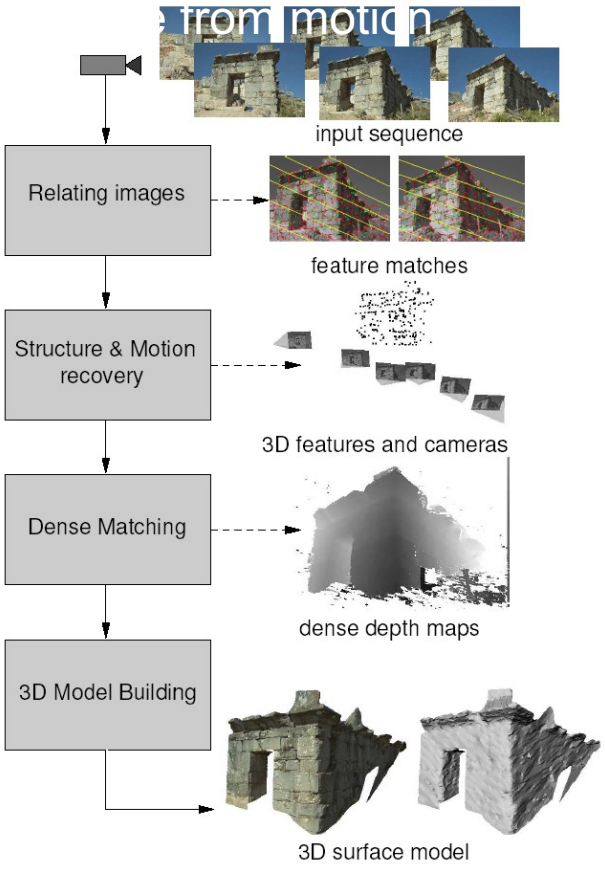


Giám sát và an ninh

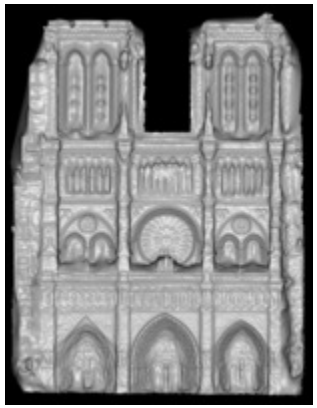


Ảnh y tế, viễn thám, thiên văn

Machine vision can be used as instrumentation

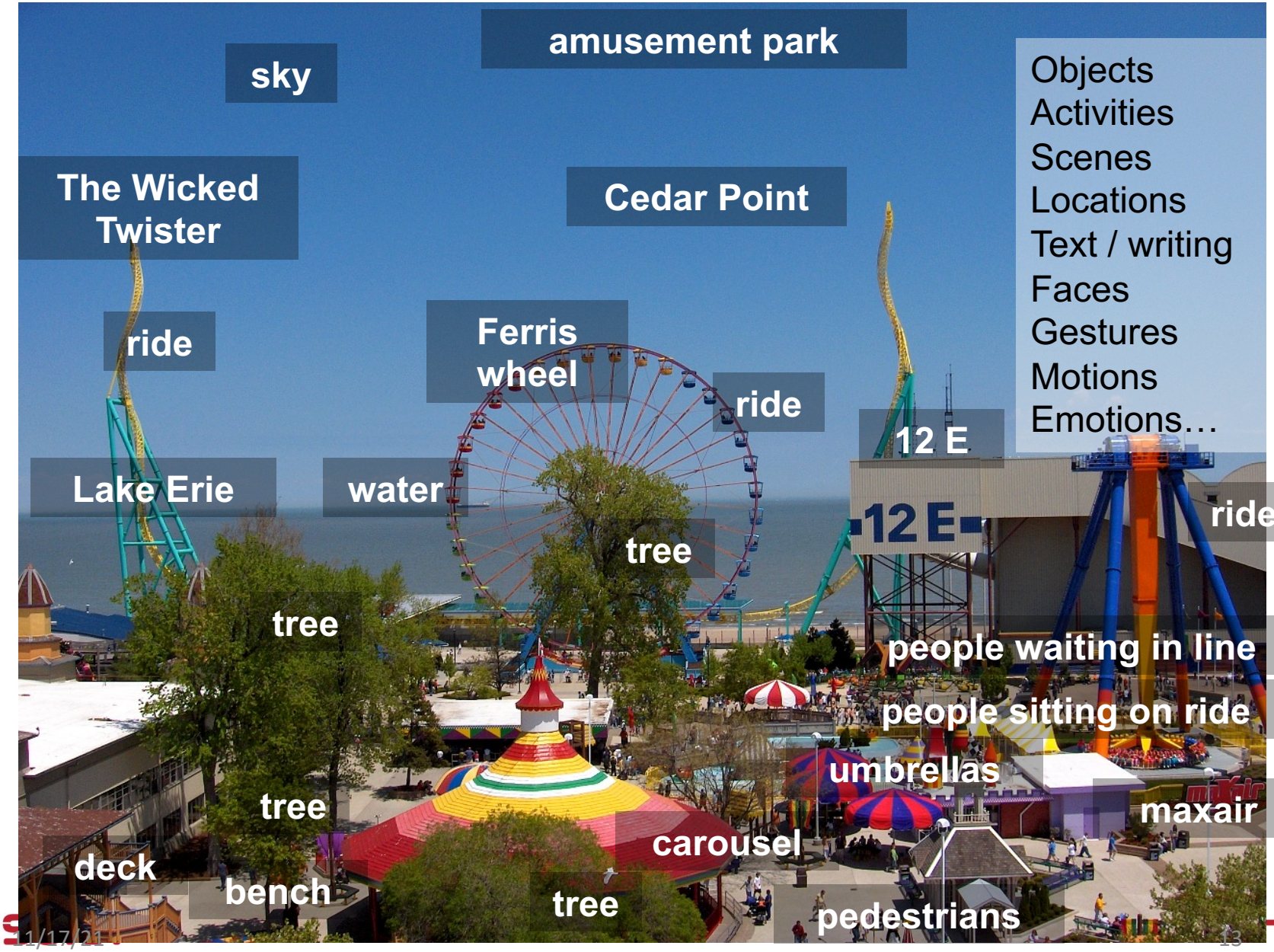


Pollefeys et al.



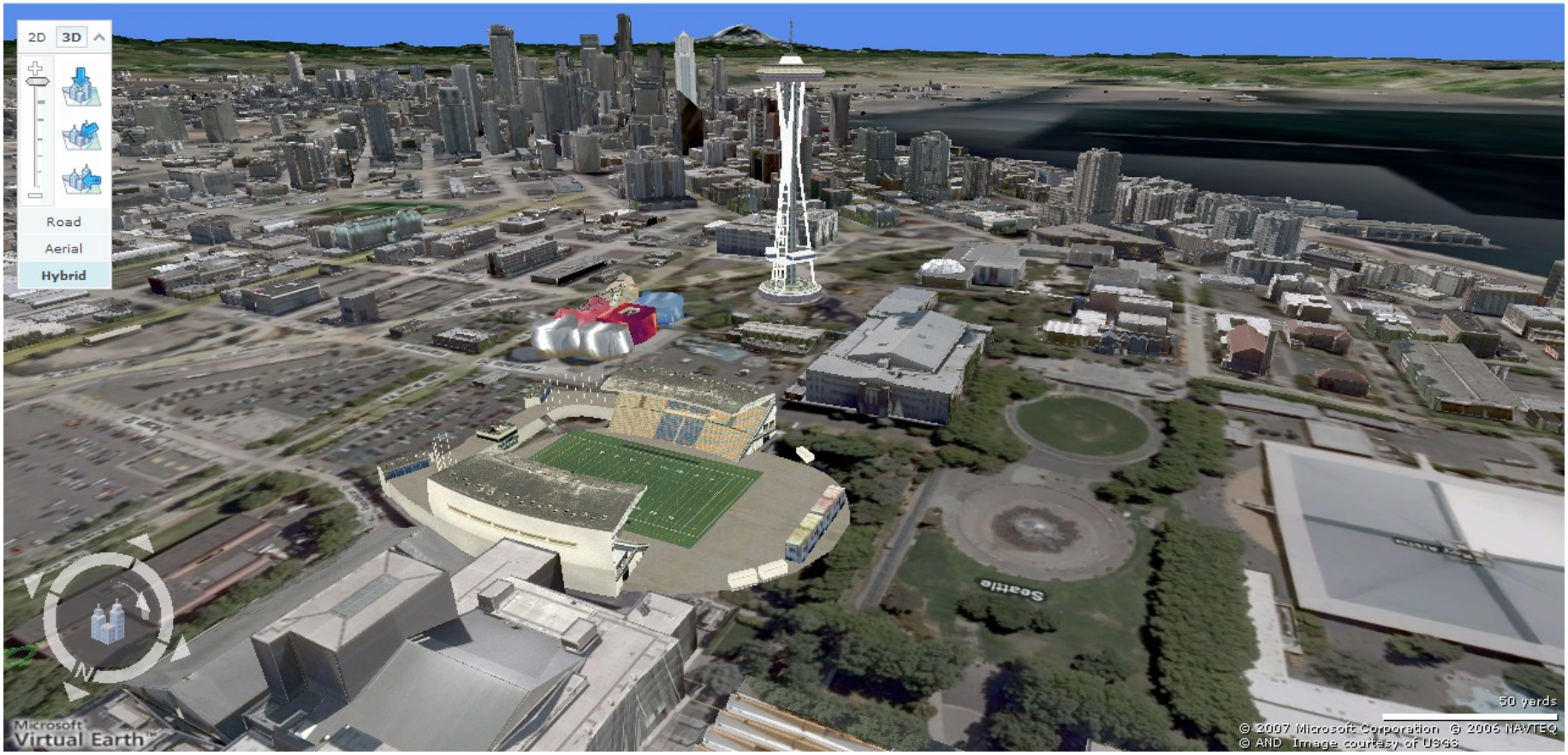
Goesele et al.

Machine vision is a source of semantic information



Slide credit: Kristen Grauman

City 3D modeling



Bing maps, Google Streetview

Source: S. Seitz

Face detection

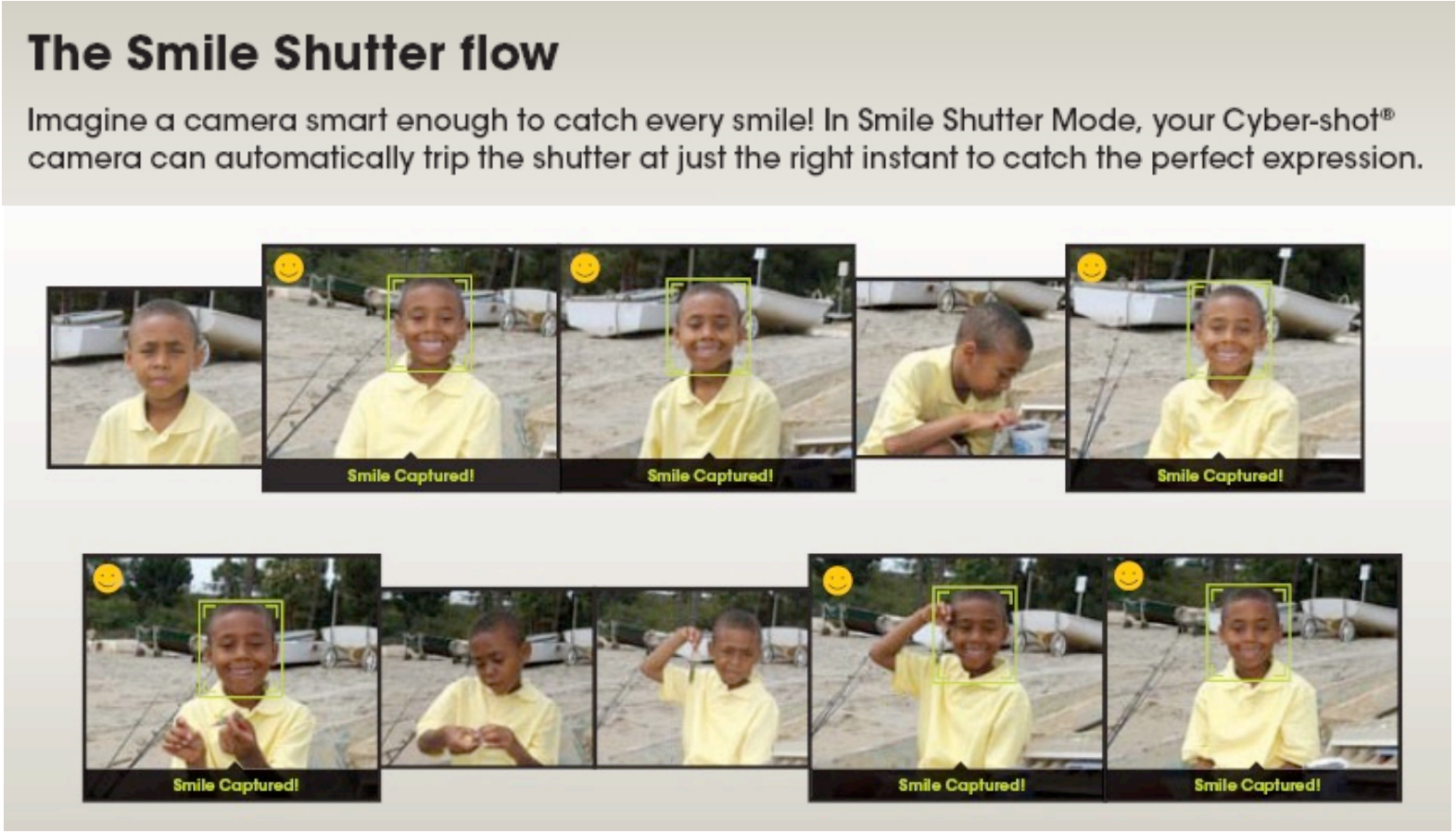
- Many digital cameras can automatically detect faces
  - Canon, Sony, Fuji, …

Source: S. Seitz

Face recognition: Apple iPhoto

http://www.apple.com/ilife/iphoto/

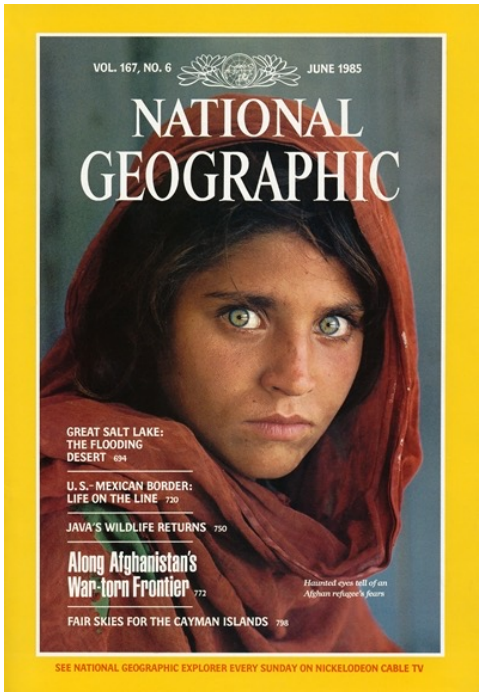
Smile detection



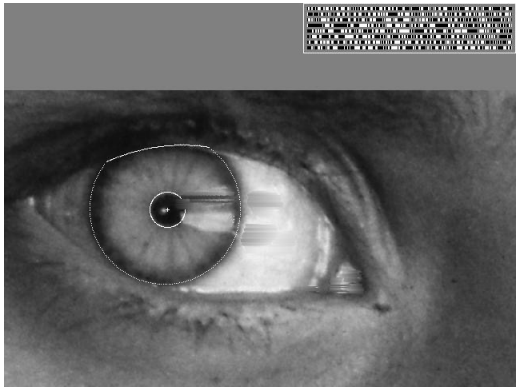
Sony Cyber-shot® T70 Digital Still Camera

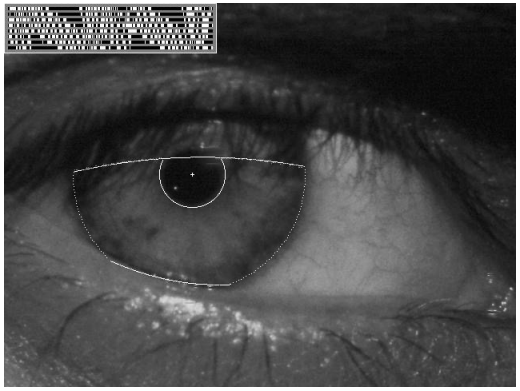
Source: S. Seitz

Biometrics



[How the Afghan Girl was Identified by Her Iris Patterns]





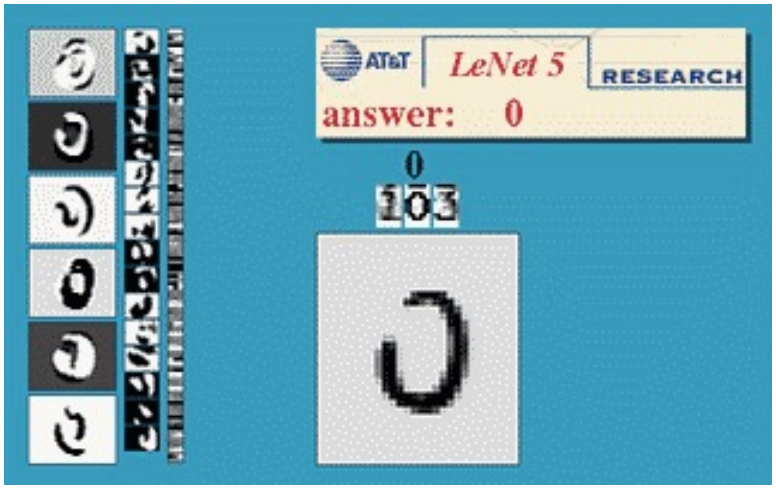
Source: S. Seitz

Biometrics

Fingerprint scanners on many laptops and devices

Facial recognition systems appear more and more
Example: iphone X just introduced faceID

Optical character recognition (OCR)



Handwritten digit recognition, AT&T labs

Plate reader
http://en.wikipedia.org/wiki/Automatic_number_plate_recognition

Source: S. Seitz

Human computer interactions and games

Microsoft's Kinect

Sony EyeToy

Assistive technologies

Source: S. Seitz

Augmented reality

Virtual reality

Applications in robotics and space exploration

NASA'S Mars Exploration Rover Spirit captured this westward view from atop a low plateau where Spirit spent the closing months of 2007.

Vision used for many different tasks:

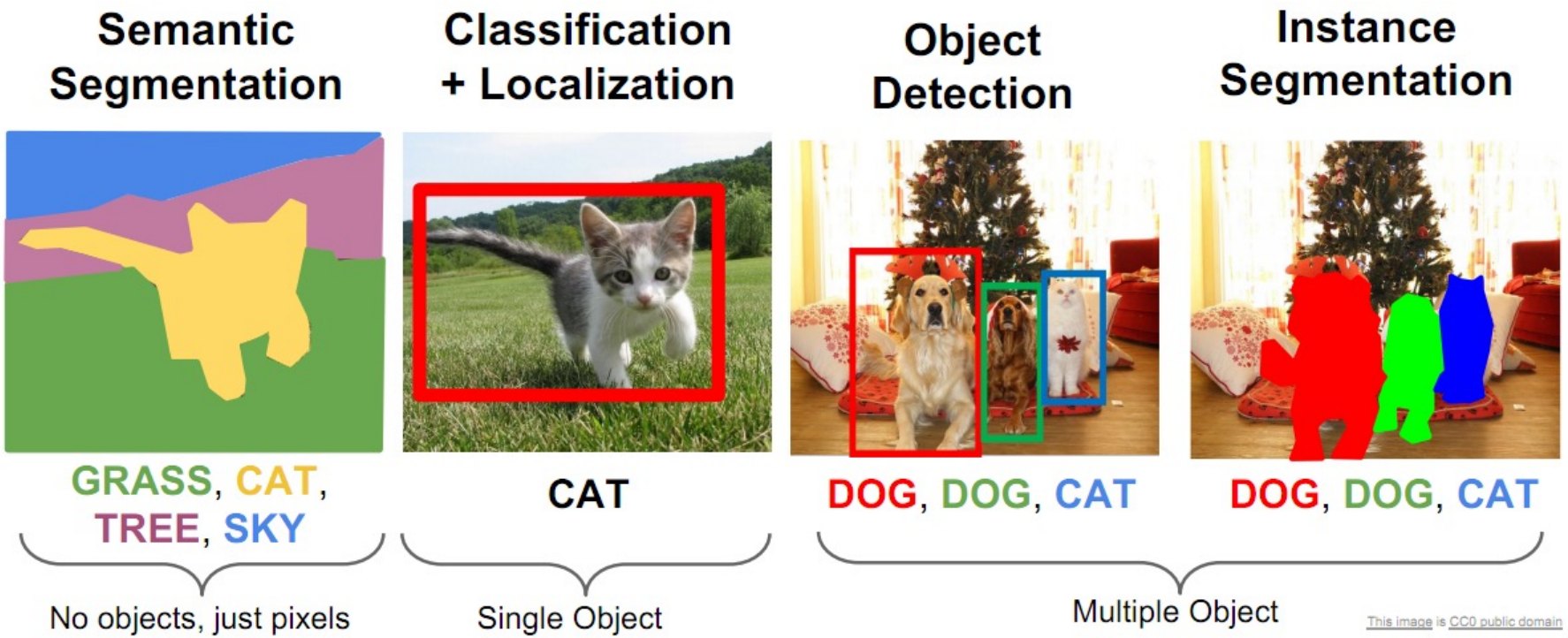
- Panorama
- 3D modeling of the Martian surface
- Obstacle detection, location tracking

For details see “Computer Vision on Mars” by Matthies et al.

Source: S. Seitz

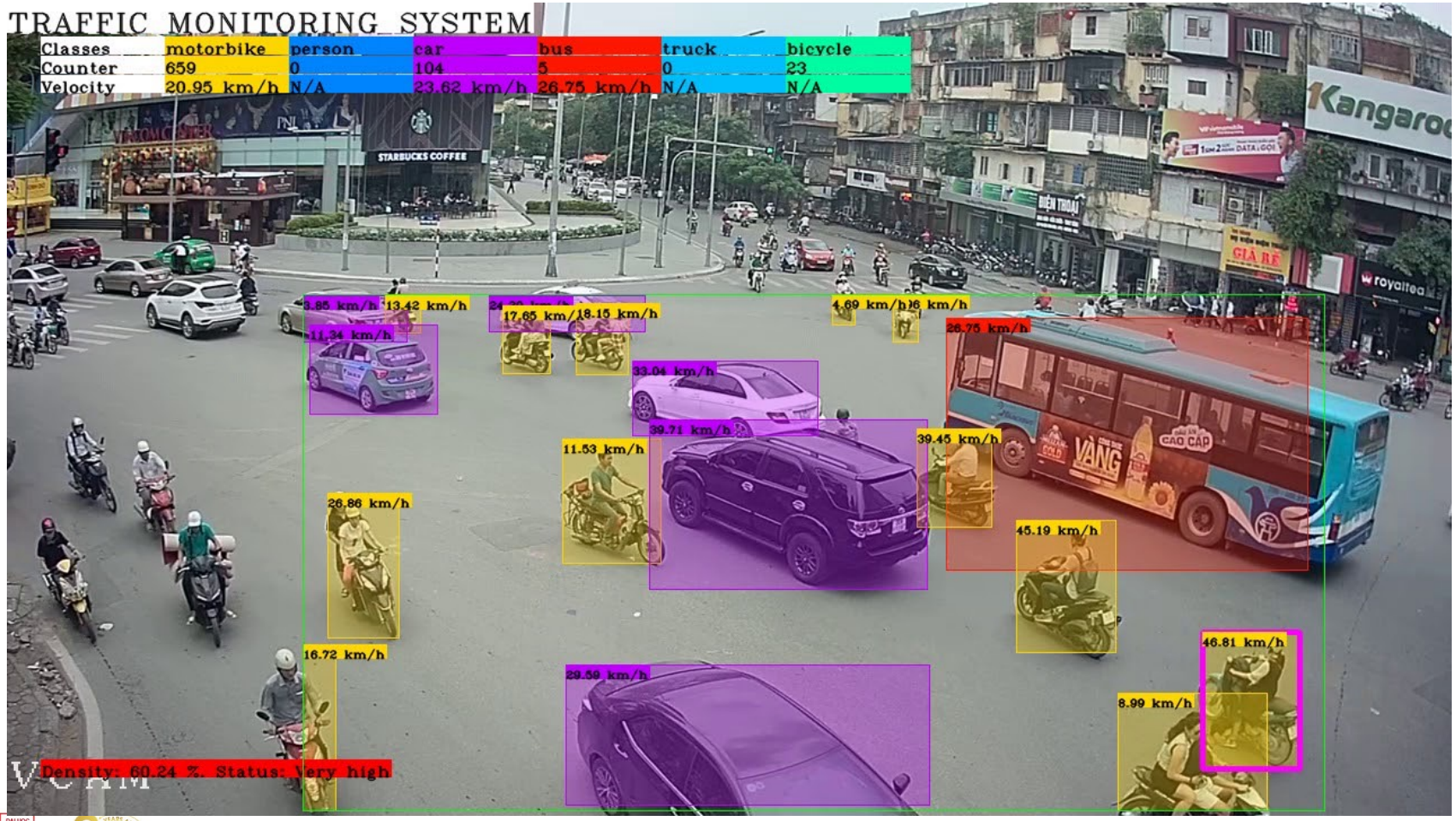
Introduction to object detection

Computer vision problems



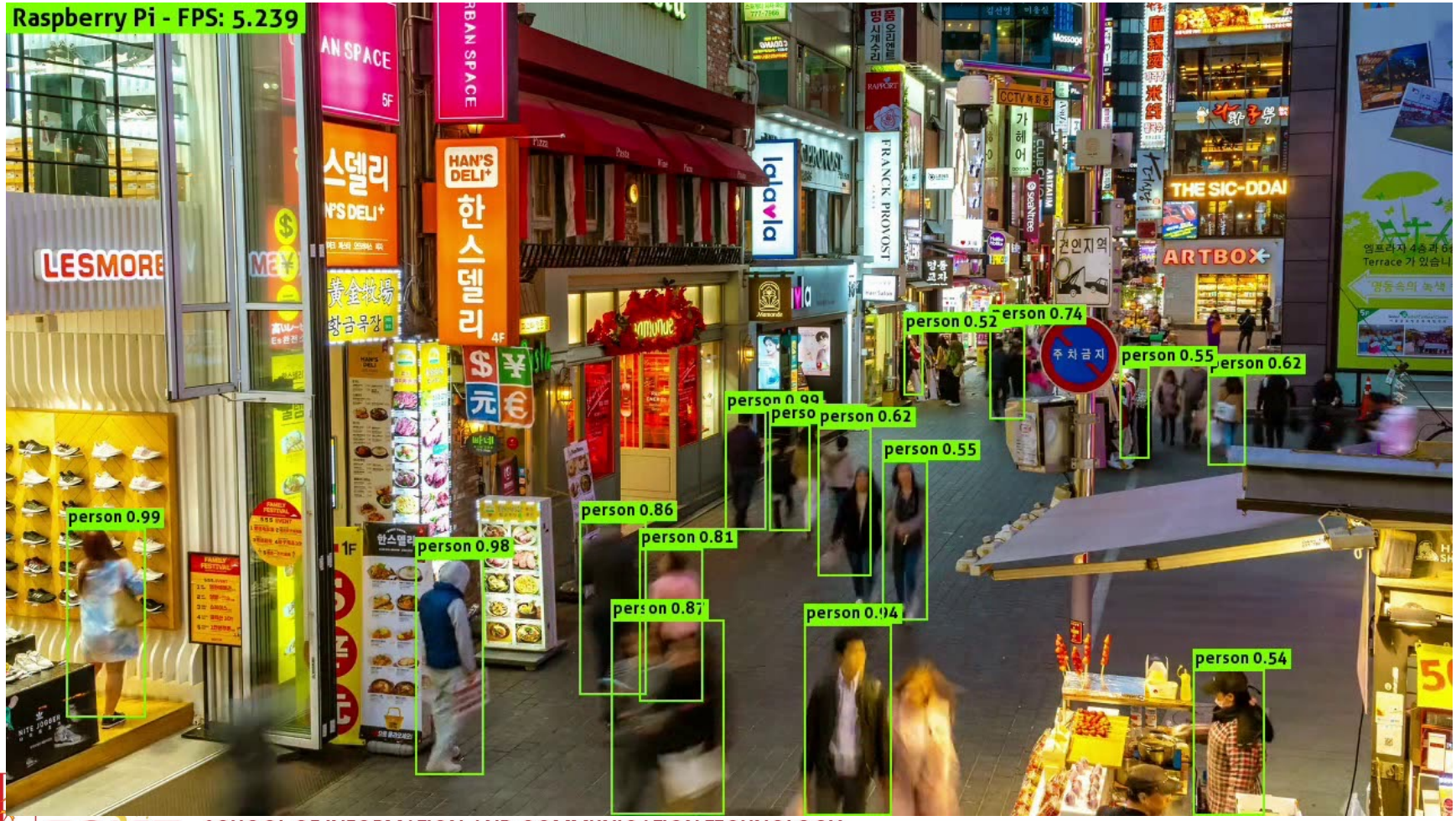
Some applications of object detection

- Smart traffic



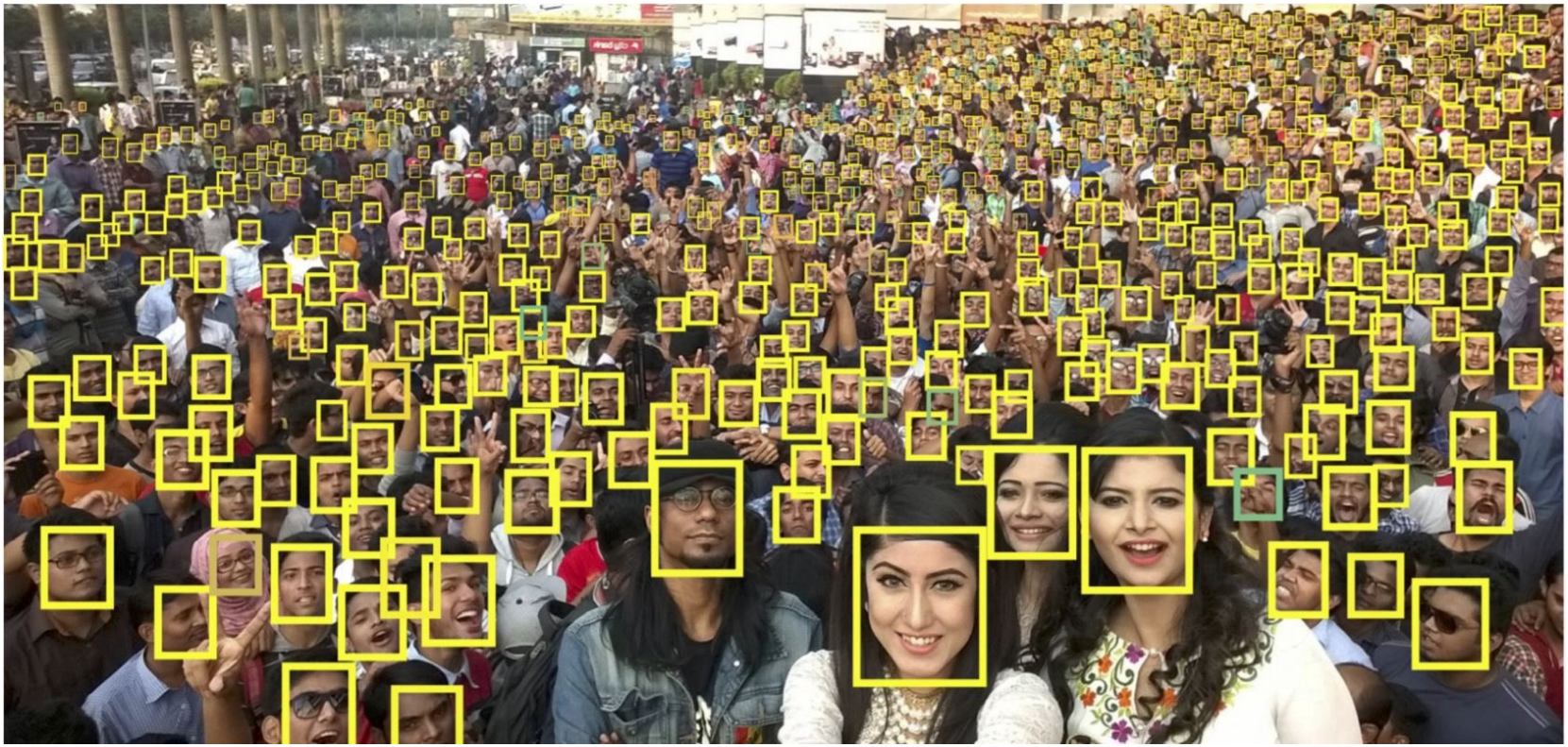
Some applications of object detection

- Human detection



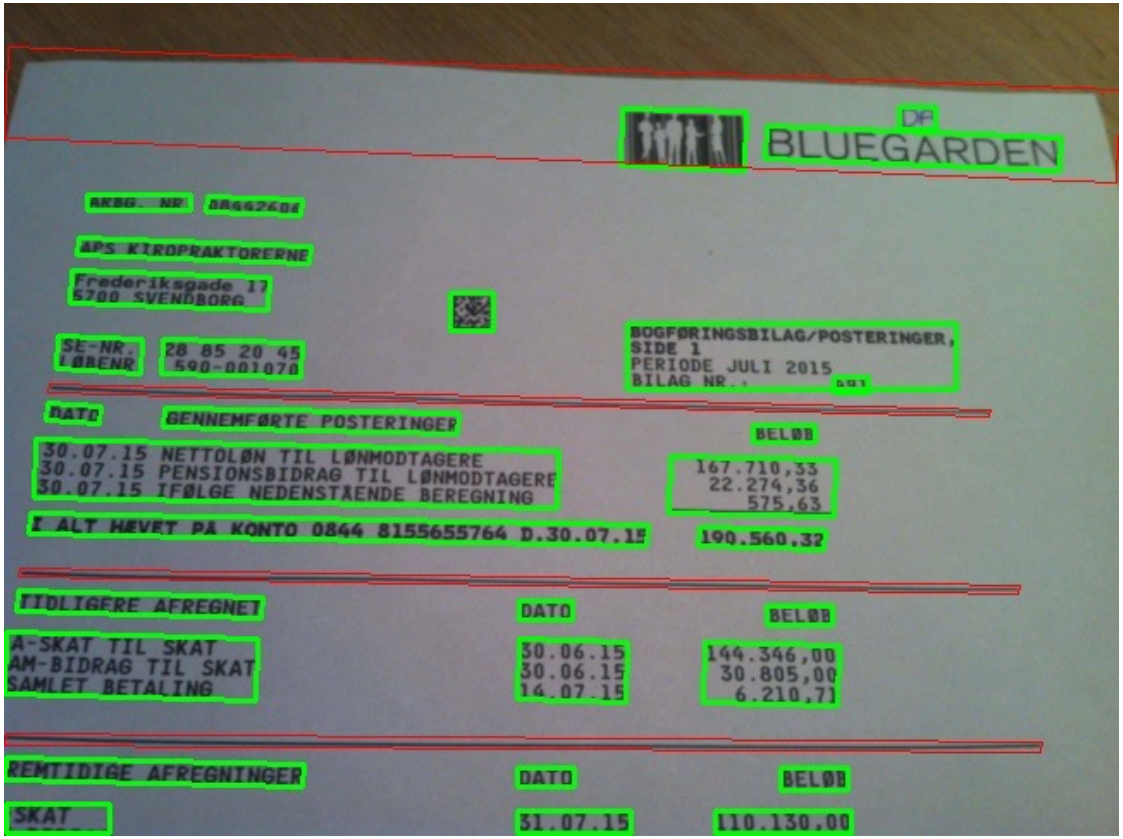
Some applications of object detection

- Face detection



Some applications of object detection

- Text detection

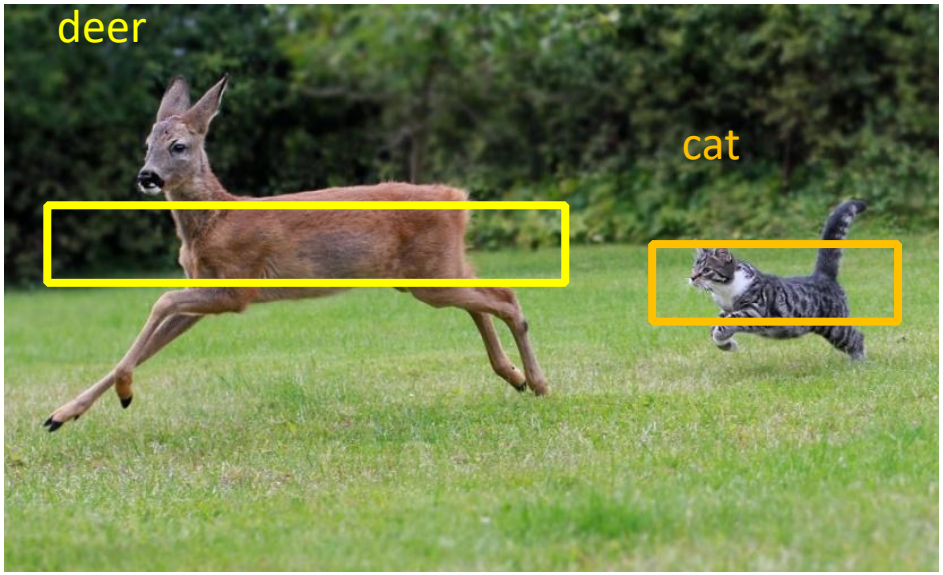


Some applications of object detection

- Automatic robot picking strawberries

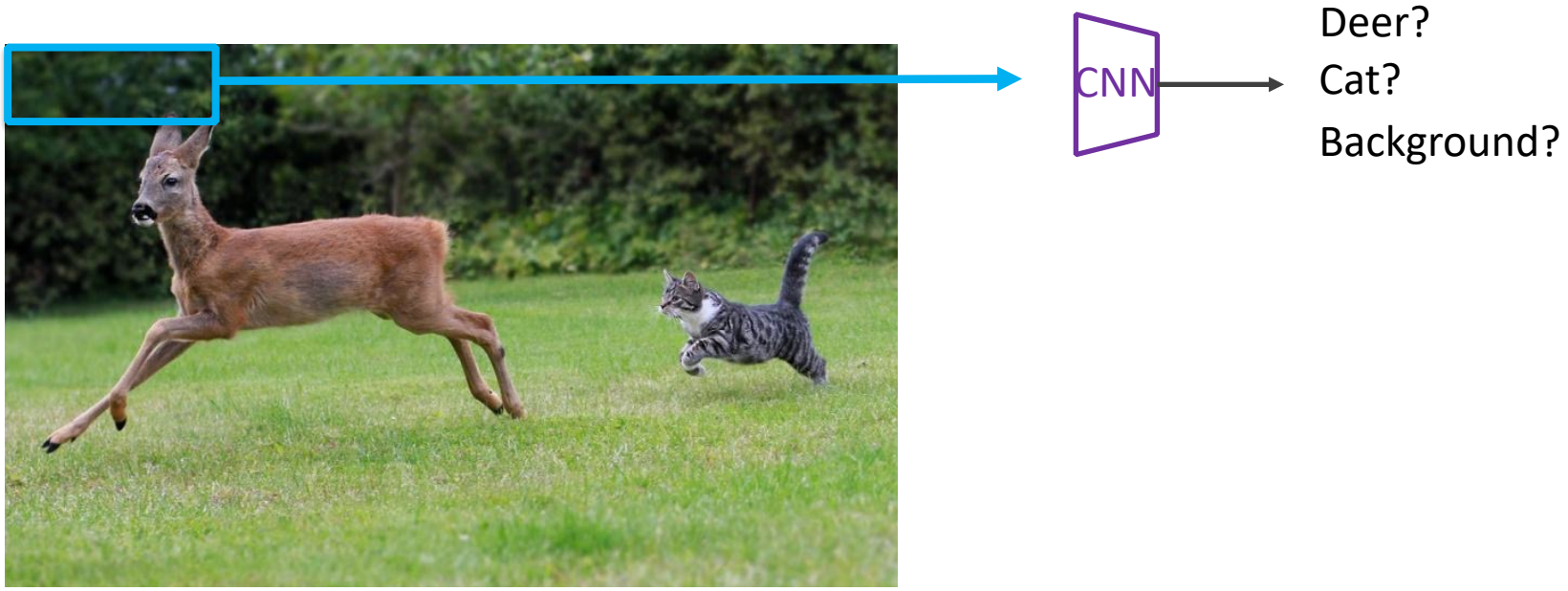
Regional proposal networks (two-stage object detectors)

Sliding window approach



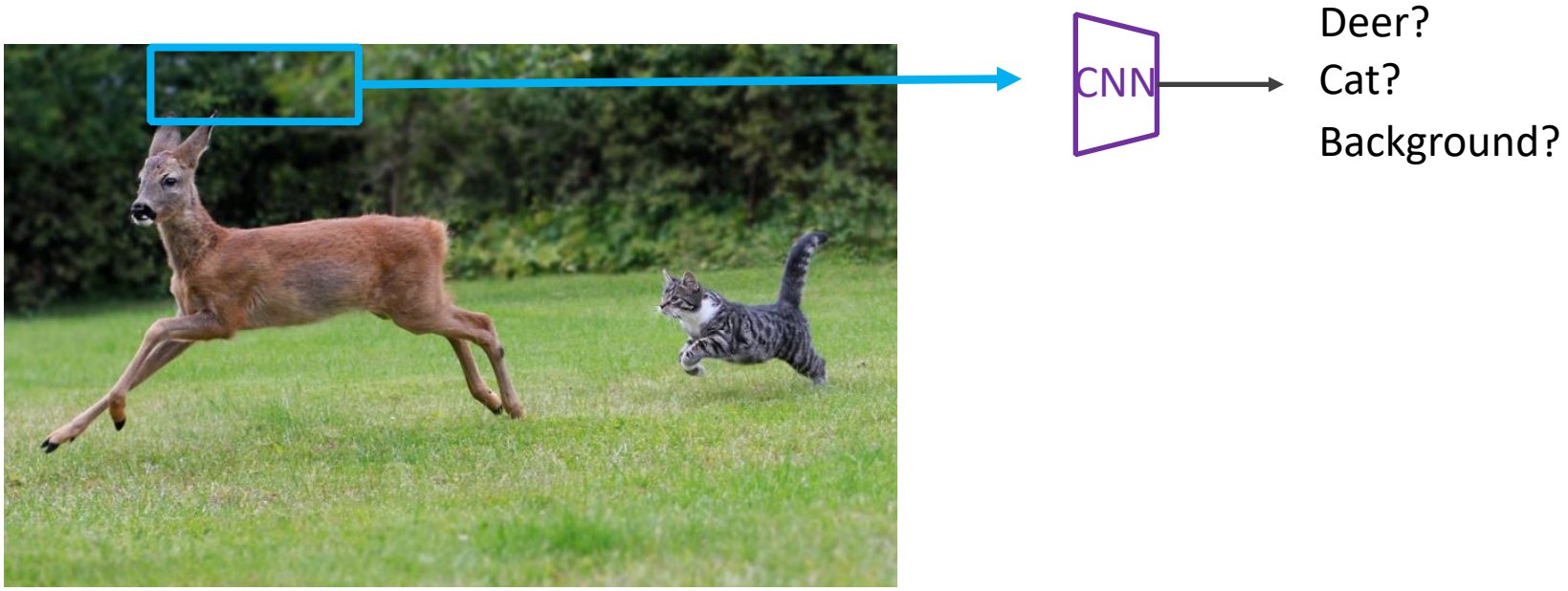
Sliding window

- Scan the window from left to right, top to bottom. At each position, the current window area is classified into different classes plus the background layer.



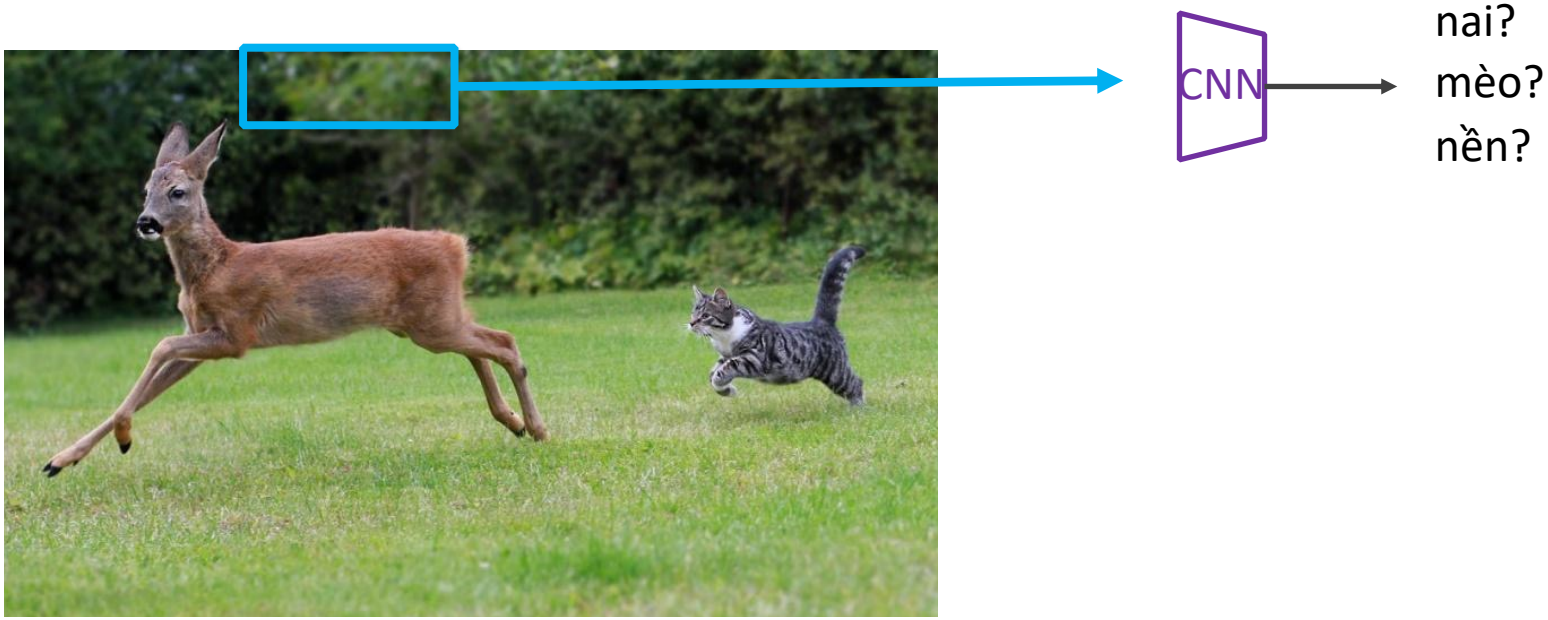
Sliding window (2)

- Scan the window from left to right, top to bottom. At each position, the current window area is classified into different classes plus the background layer.



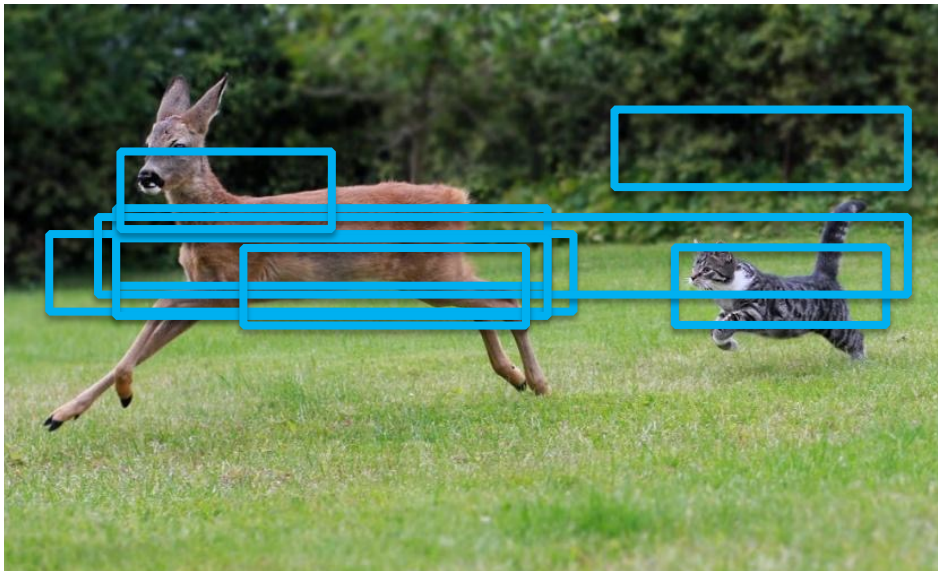
Sliding window (3)

- Scan the window from left to right, top to bottom. At each position, the current window area is classified into different classes plus the background layer.



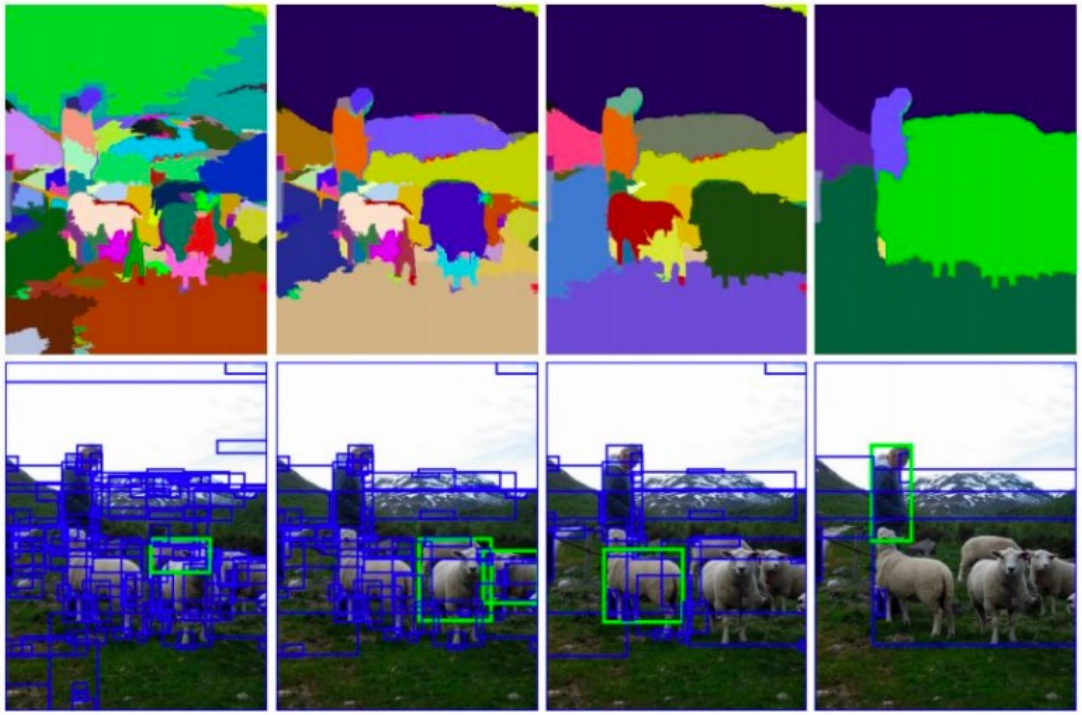
Regional proposal approach

- Instead of scanning all locations (very large numbers!), analyze to only suggest some areas (boxes) with a high probability of containing the object.
- These methods have two stages:
  1. region proposal
  2. process each region to classify and to correct box . coordinates



SS: Selective Search

- Segmentation As Selective Search for Object Recognition. van de Sande et al. ICCV 2011

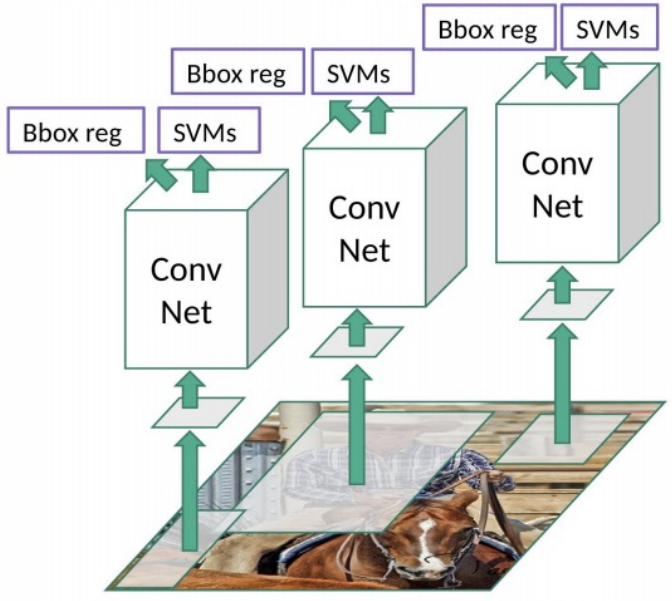


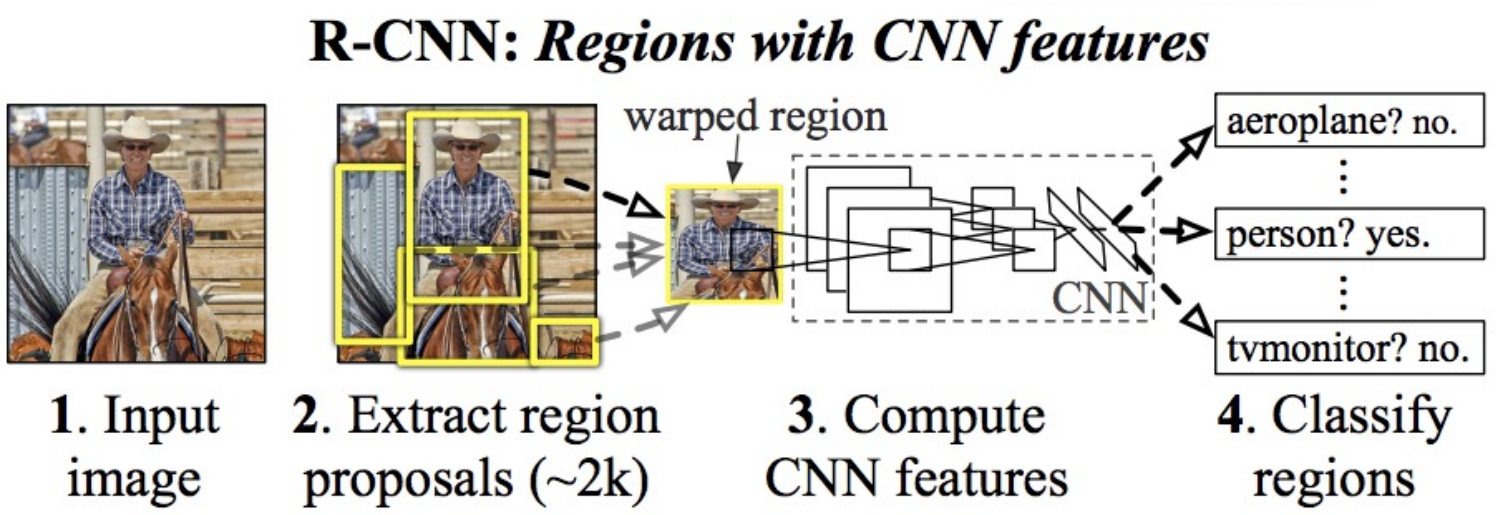
Fast with very high recall
Over-segmenting the image based on intensity of the pixels

1. Add all bounding boxes corresponding to segmented parts to the list of regional proposals
2. Group adjacent segments based on similarity
3. Go to step 1

R-CNN (Region-based ConvNet)

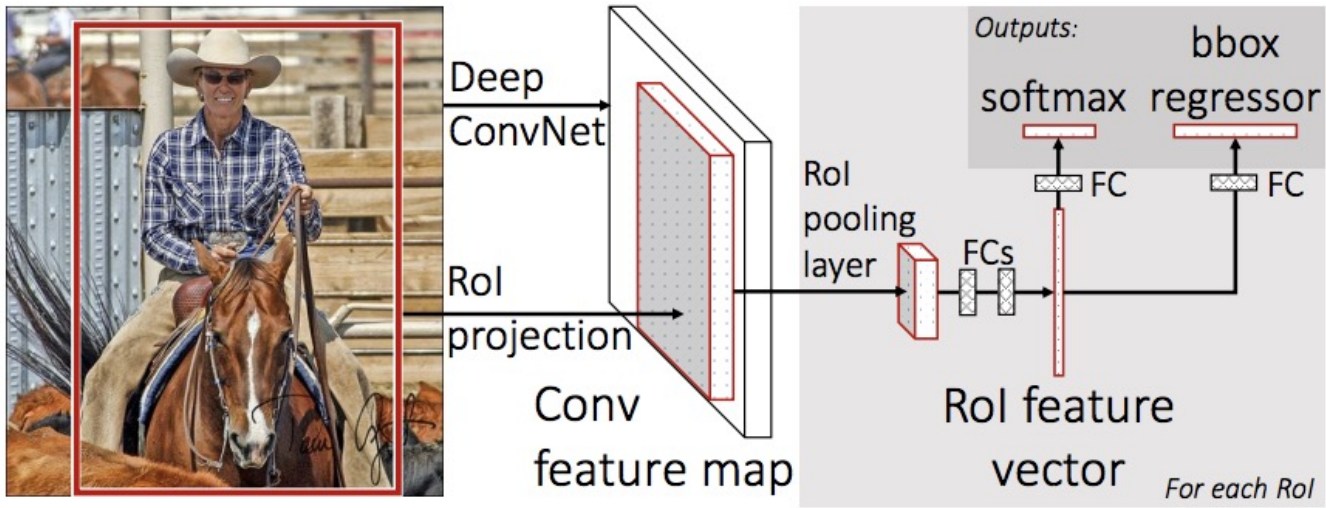
- Suggest some potential regions using other raw algorithms, such as selective search
- Using CNN to extract features of each region and then classify by SVM





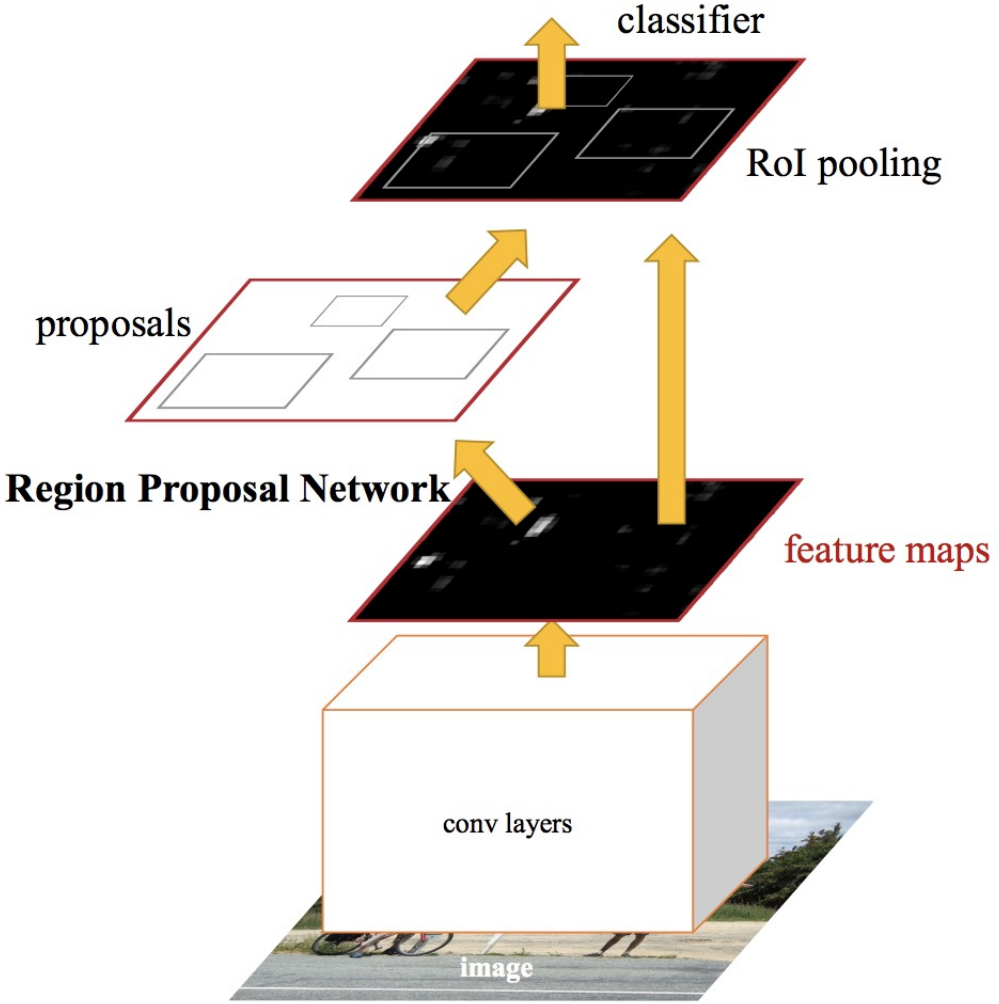
Fast-RCNN

- Push all regions (about 2000) over CNN extractor network at the same time
- Crop the information in the output layer of the CNN instead of cropping the area on the original image like R-CNN
- Push through the classification branch and the bbox regressor branch



Faster-RCNN

- Use a dedicated network to recommend regions instead of selective search
- Also called two-stage object detector method.



None regional proposal networks (one-stage object detectors)

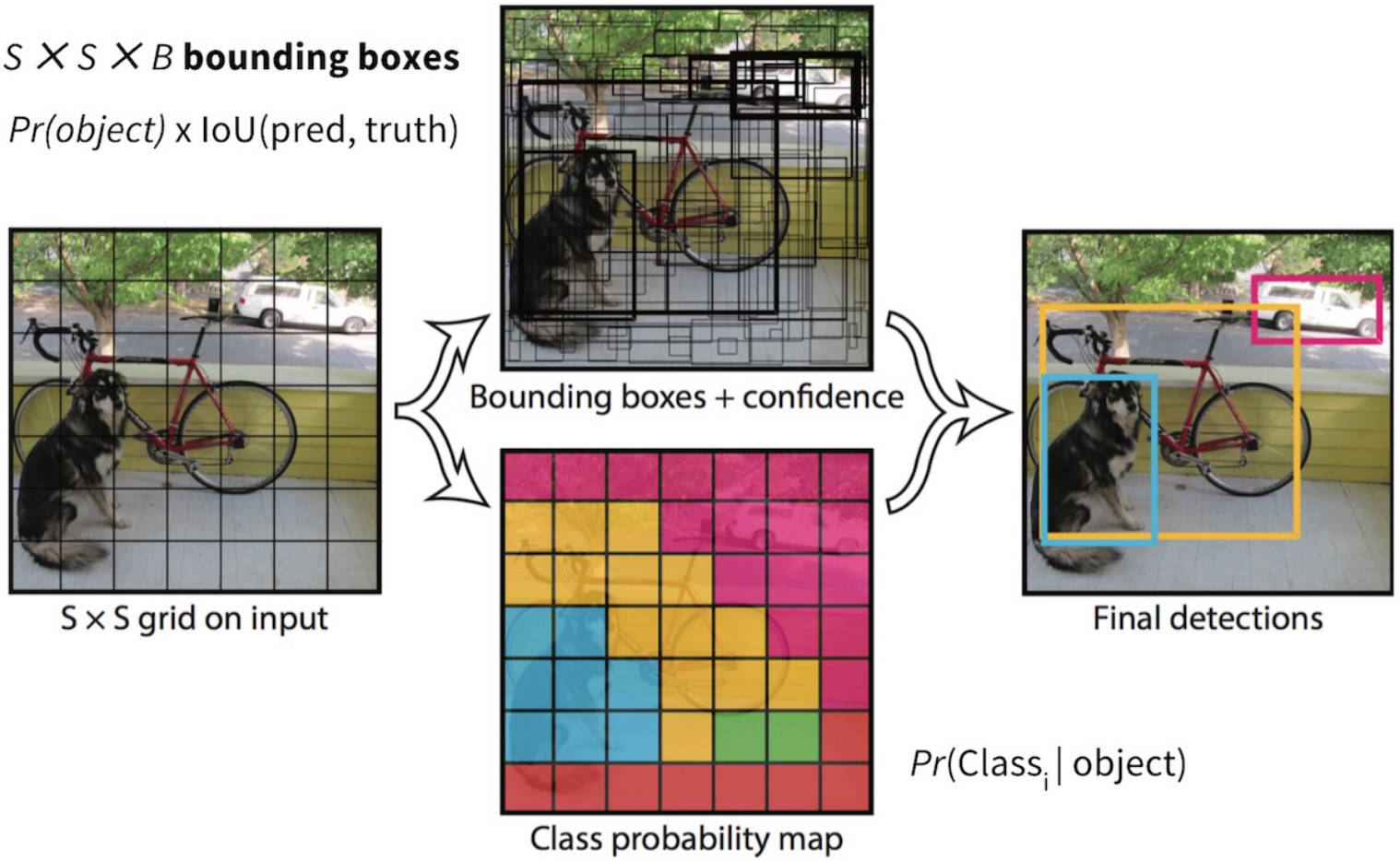
One-Stage Object Detectors

- These networks often suggest a dense box grid on the original image, often with a consistent stride.
- Each of these boxes will be classified and corrected for coordinates (if the box contains objects) using the CNN network
- One-stage networks are generally faster and simpler than two-stage networks, but the accuracy may not be as high.

YOLO- You Only Look Once

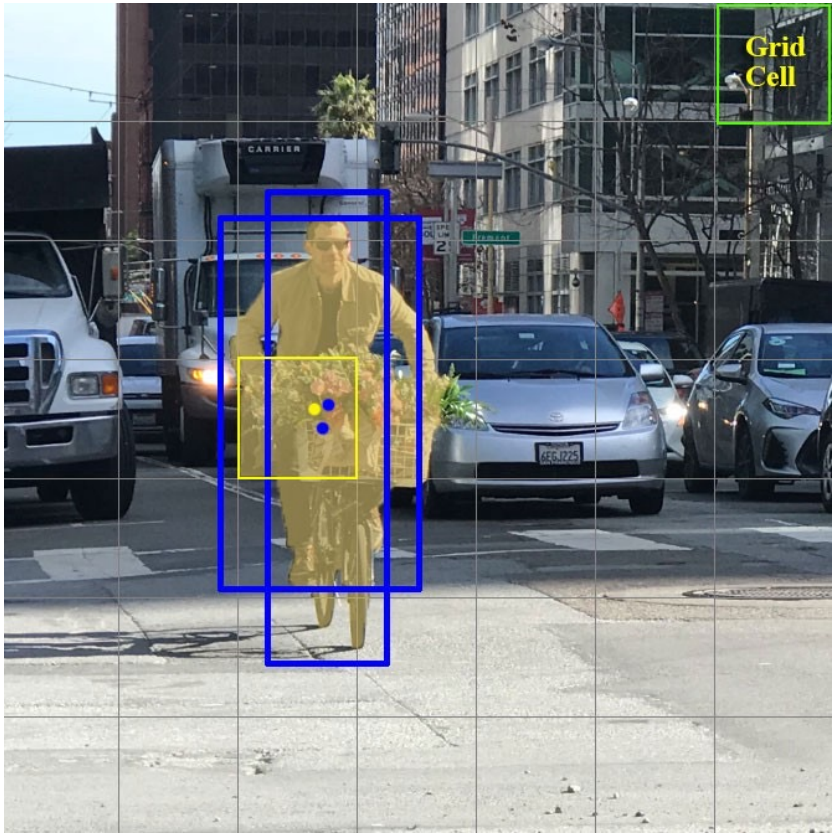
$S \times S \times B$ **bounding boxes**

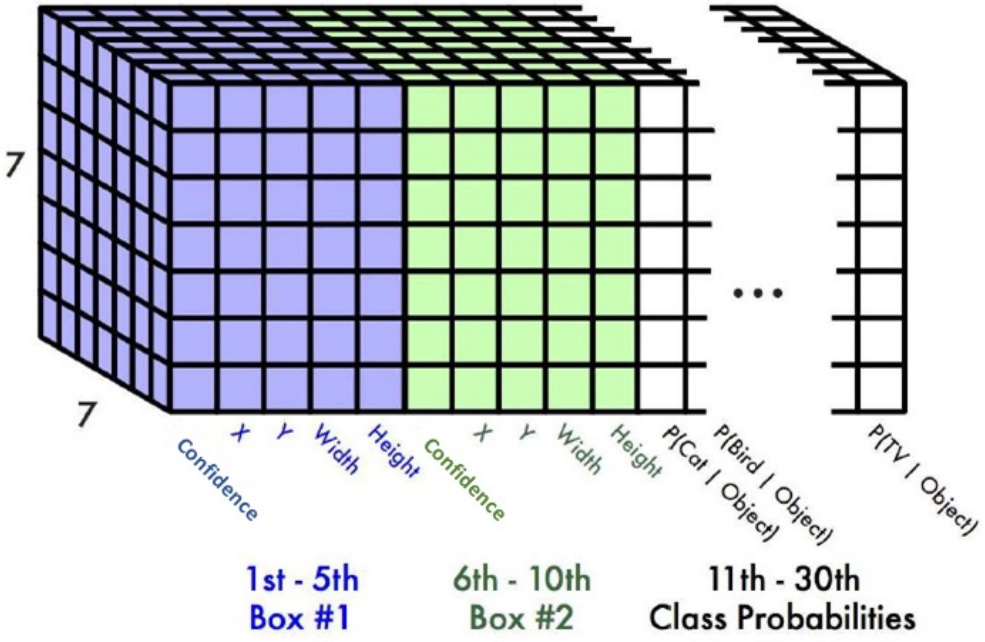
confidence = *Pr(object)* x IoU(pred, truth)



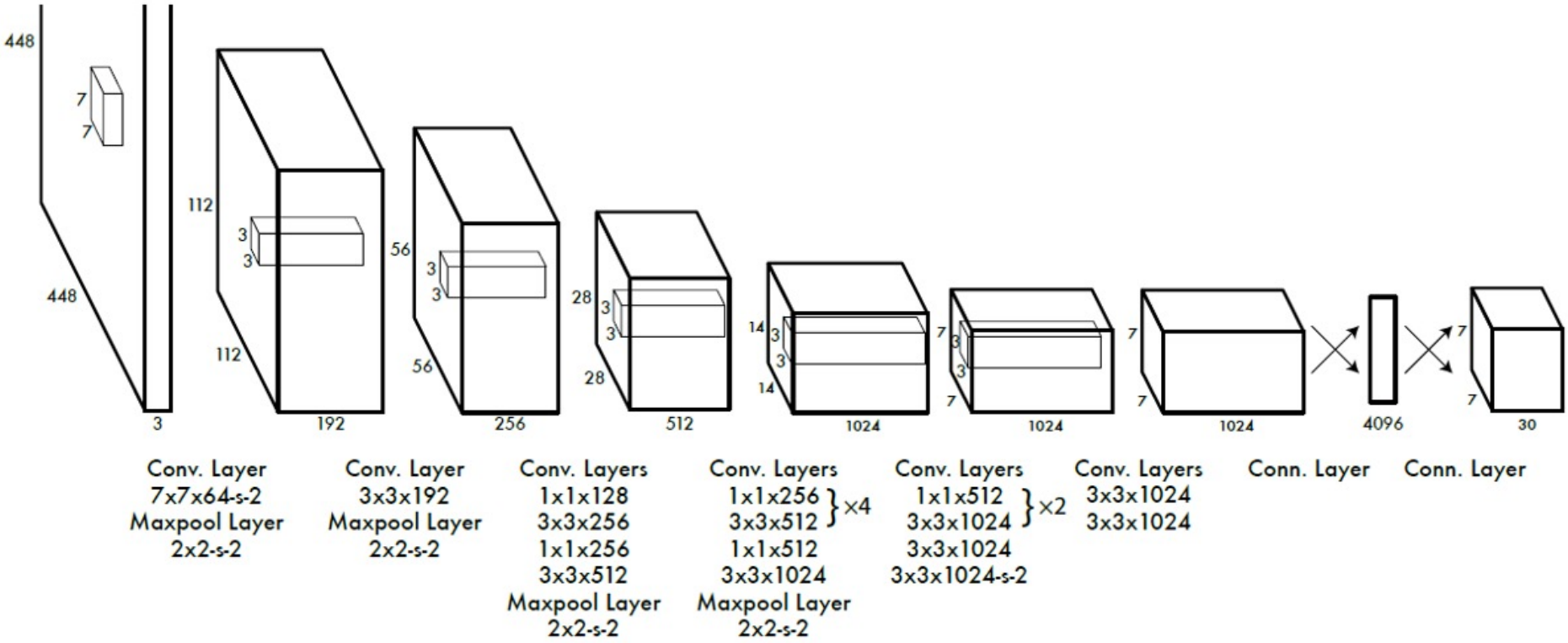
$$\Pr(\text{Class}_i|\text{Object}) * \Pr(\text{Object}) * \text{IOU}^{\text{truth}}_{\text{pred}} = \Pr(\text{Class}_i) * \text{IOU}^{\text{truth}}_{\text{pred}}$$

YOLO- You Only Look Once

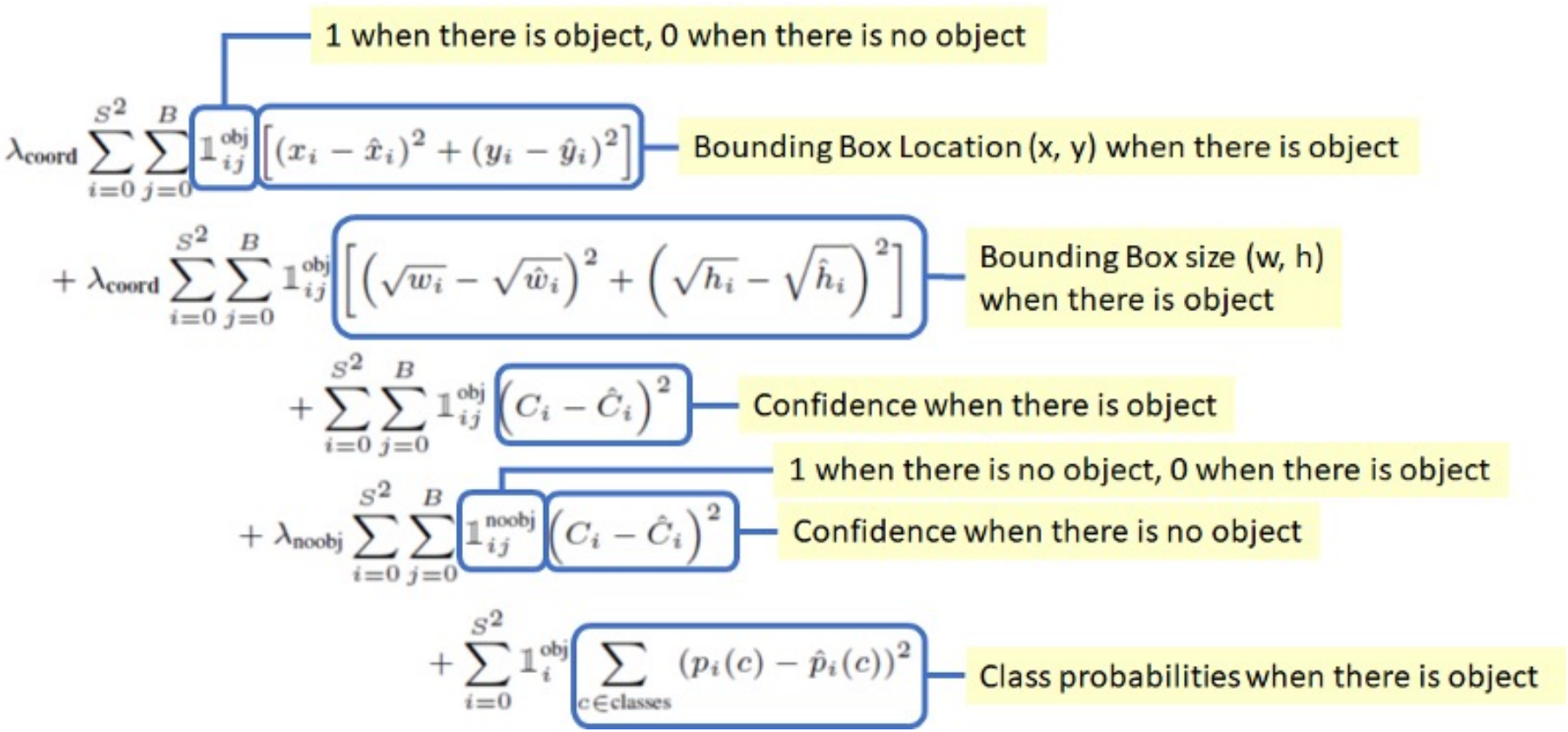




YOLO- You Only Look Once

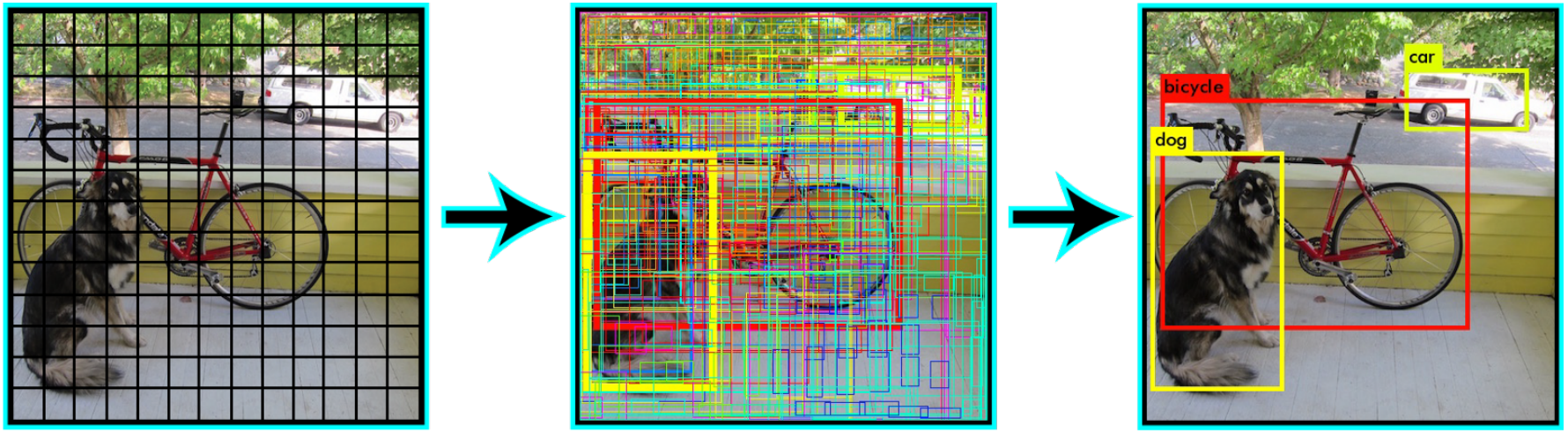


YOLO- You Only Look Once



YOLO- You Only Look Once

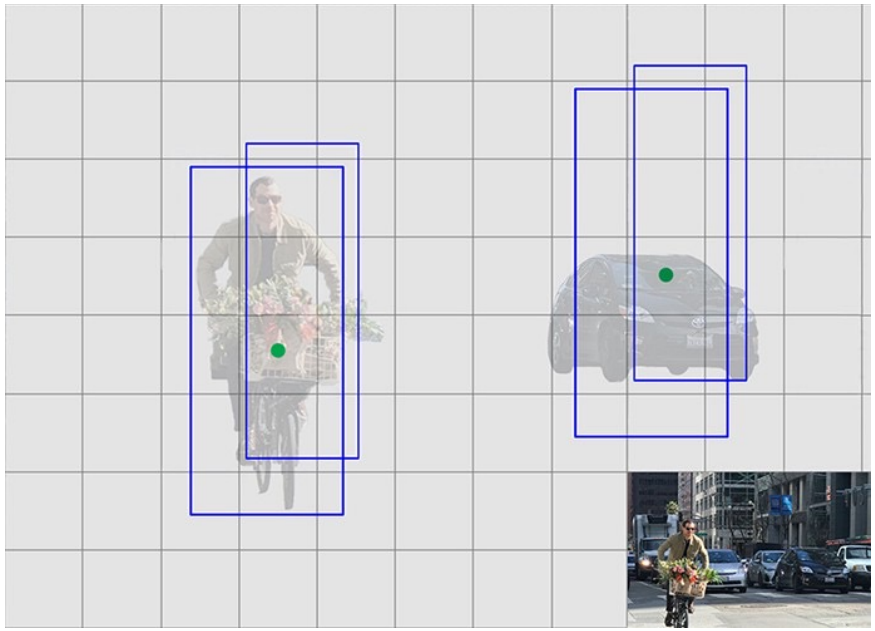
- Non-maximal suppression: gather the boxes to give the final result

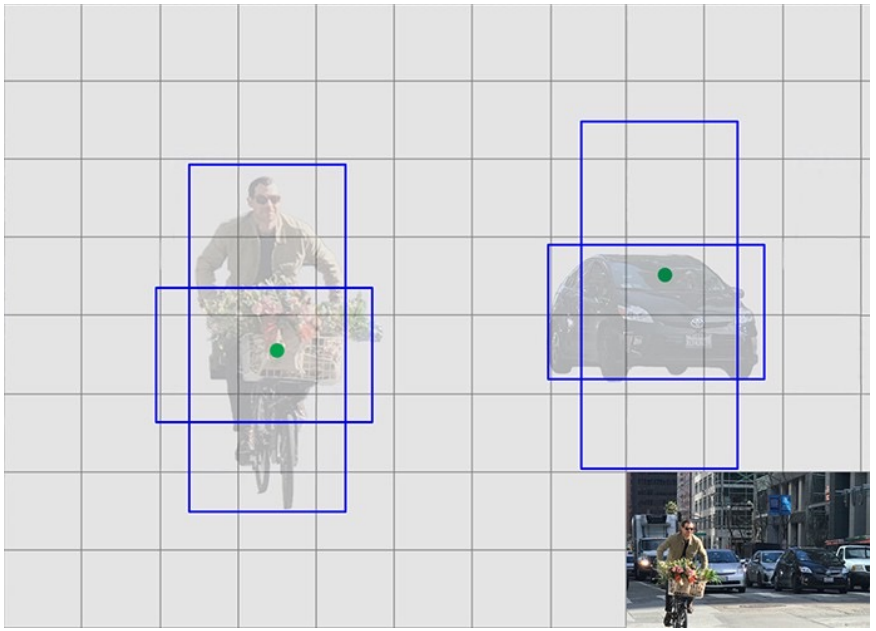


YOLO can make duplicate detections for the same object.
YOLO applies non-maximal suppression to remove duplications with lower confidence

- Sort the predictions by the confidence scores.
- Start from the top scores, ignore any current prediction if we find any previous predictions that have the same class and IoU > 0.5 with the current prediction.
- Repeat step 2 until all predictions are checked.

YOLO v2



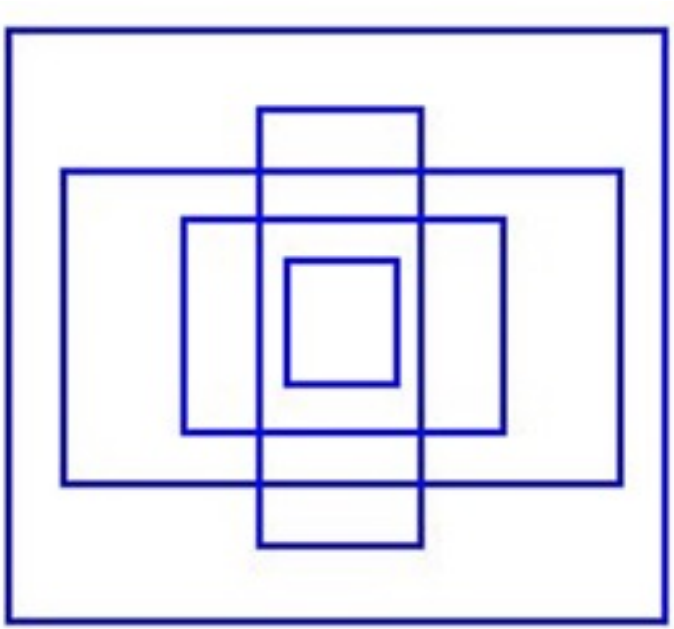


Initially, YOLO makes arbitrary guesses on the boundary boxes.
In the real-life domain, the boundary boxes are not arbitrary.
Idea

- We can create 5 **anchor** boxes with the intended shapes.
- We predict offsets to each of the anchor boxes above.
- If we **constrain** the offset values, we can maintain the diversity of the predictions and have each prediction focuses on a specific shape

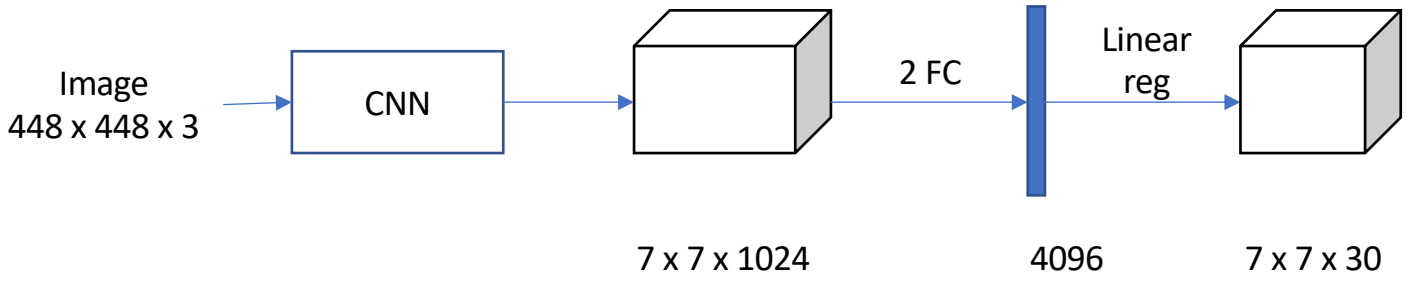
YOLO v2

- Each cell has 5 anchor boxes. For each network anchor will give the following information:
  - box offset: 4 real numbers in the range [0, 1]
  - Confidence that the box is likely to contain the object (objectness score).
  - Probability distribution of objects in that box for different classes (class scores).
- In total, each cell has an output number: 5 * (4 + 1 + 20) = 125 real numbers

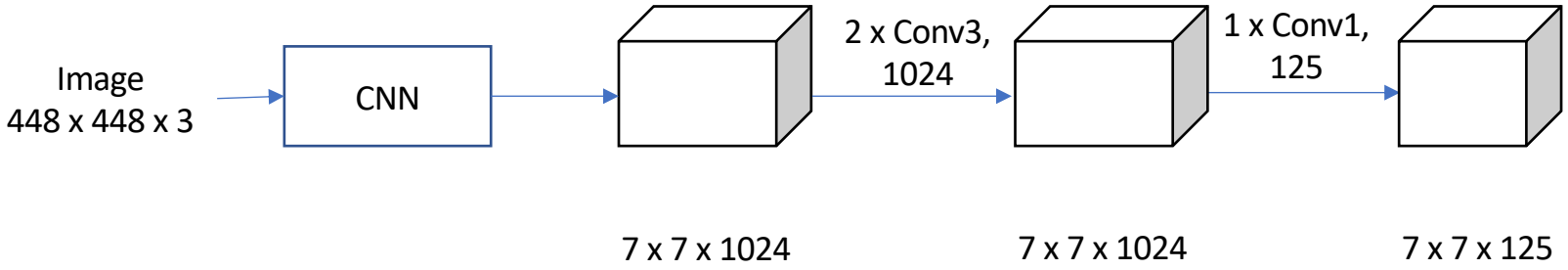


5 anchor box

YOLO v2



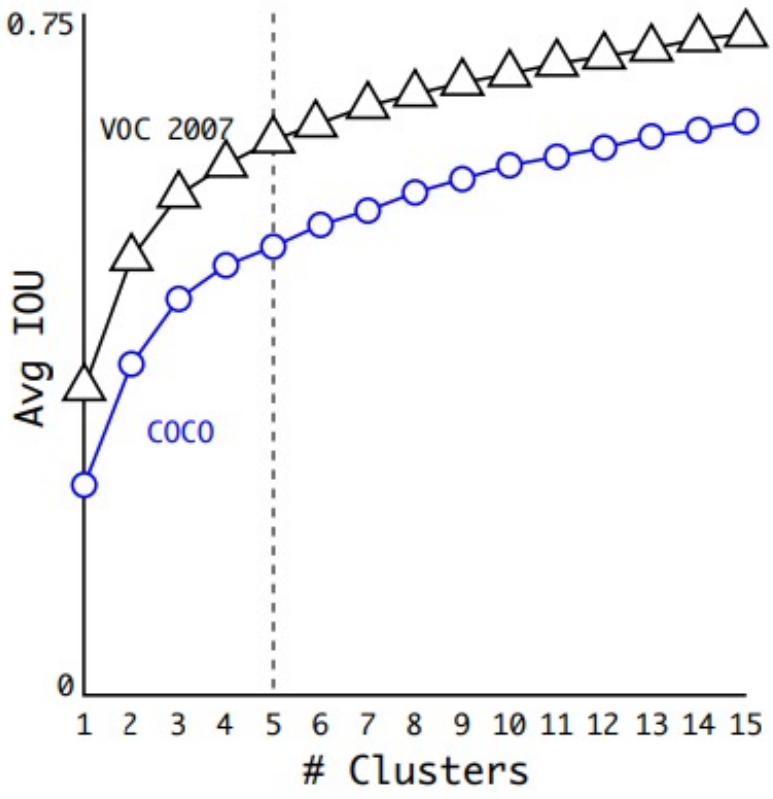
YOLO v1

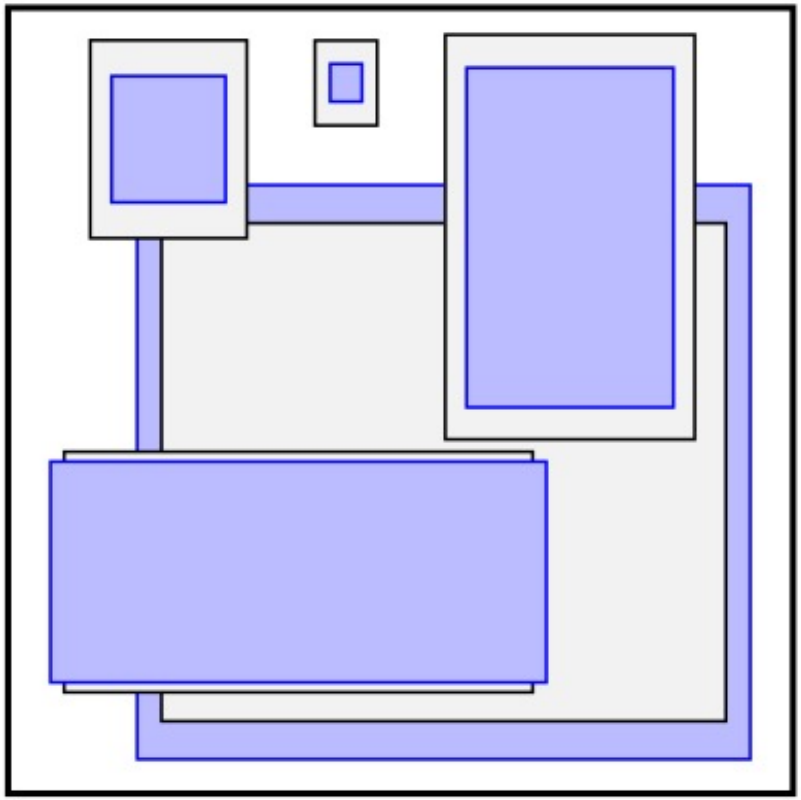


YOLO v2

YOLO v2

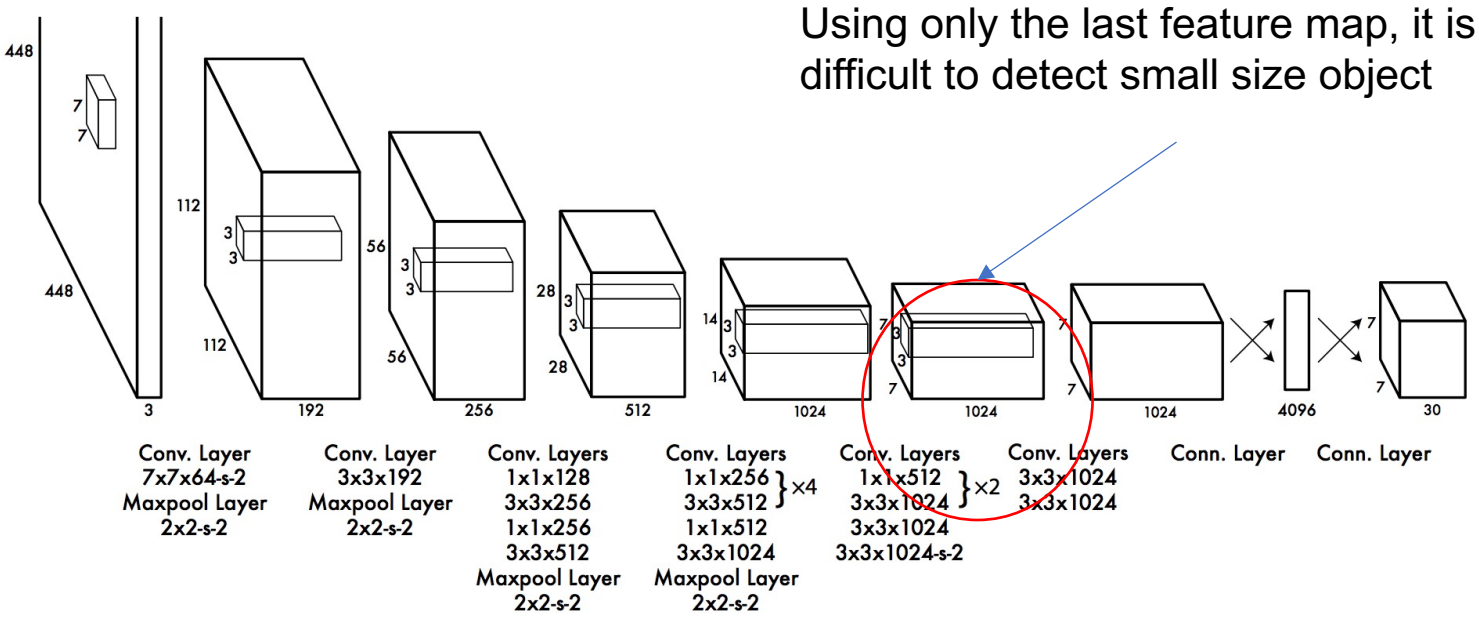
- Determine the default size of the anchors by applying k-means on the box set of labeled objects in the training set.





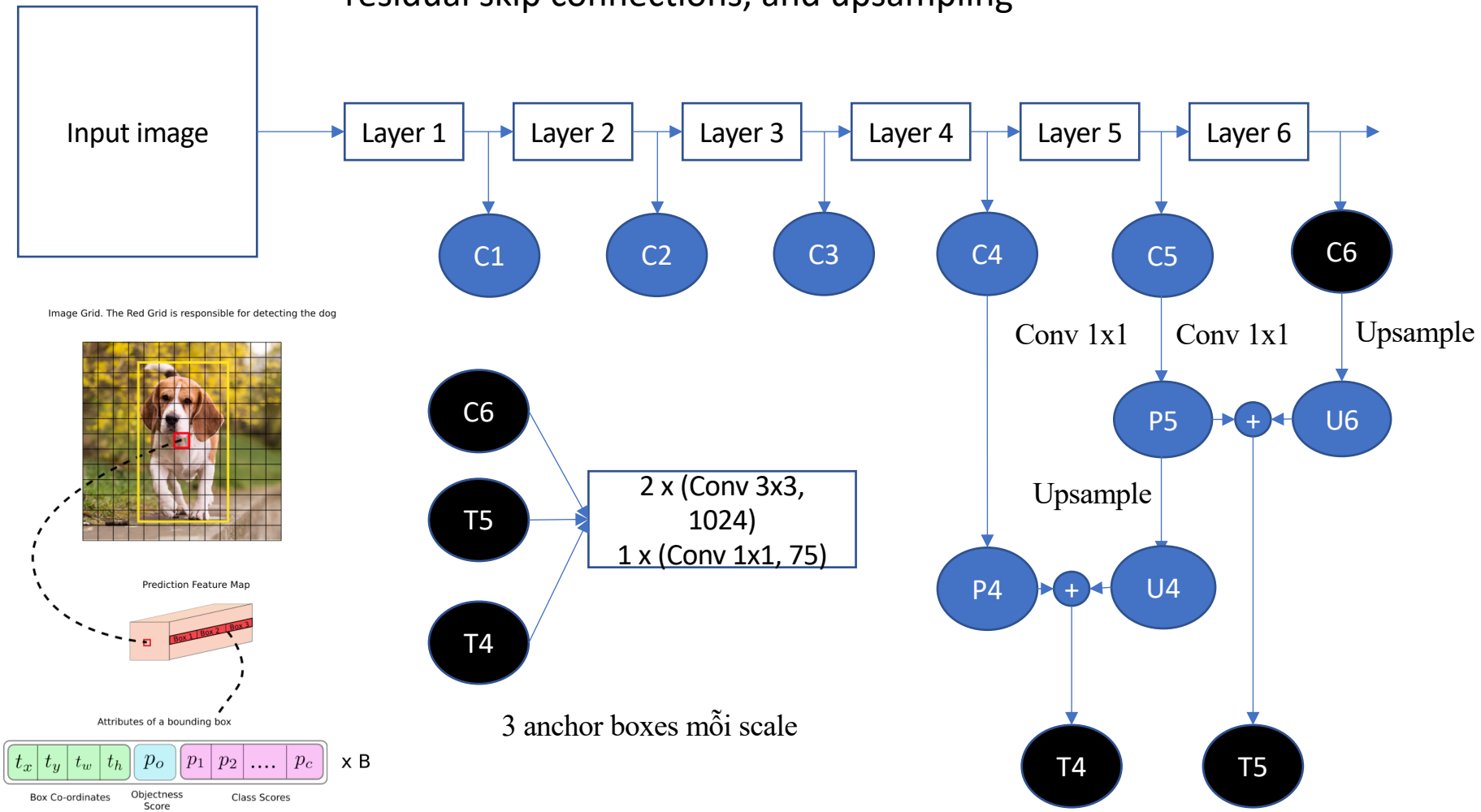
YOLO v2

- Cons of YOLO v1 and v2:

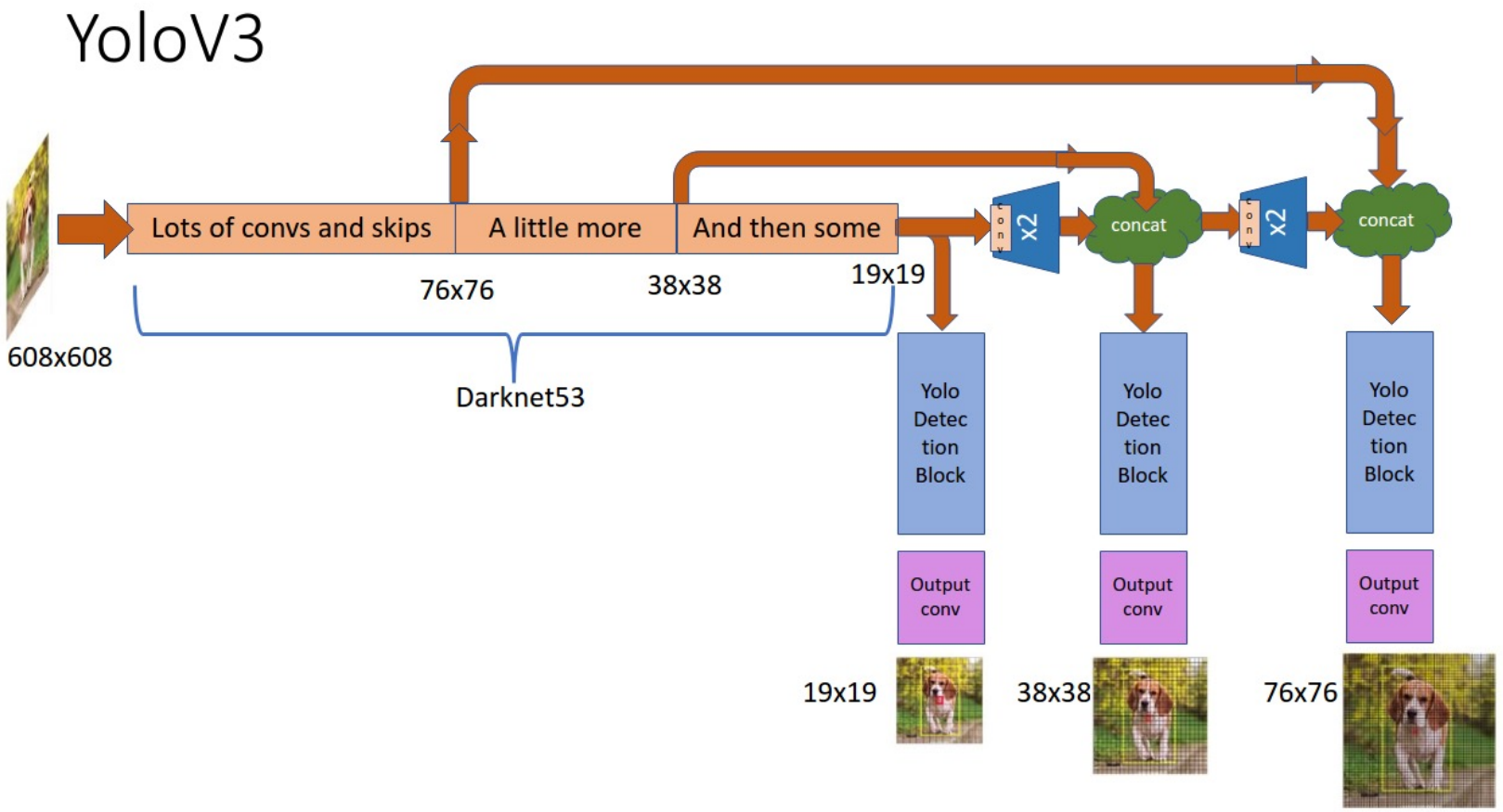


YOLO v3

residual skip connections, and upsampling

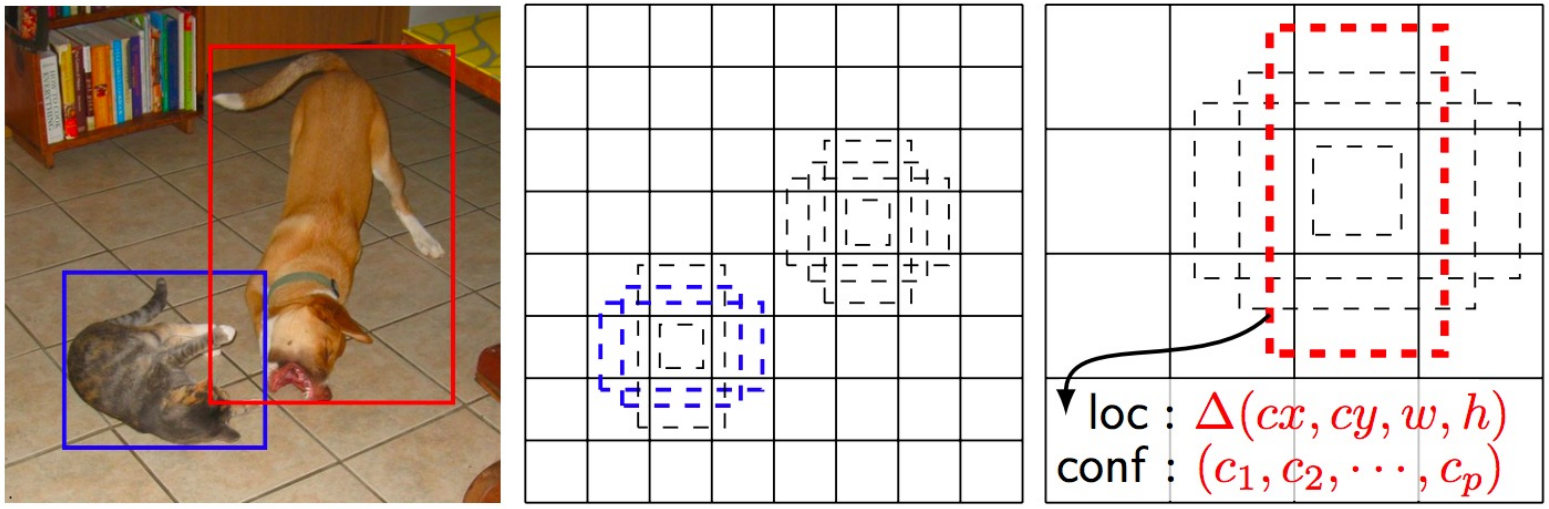


YOLO v3


YoloV3
608x608
Lots of convs and skips
A little more
And then some
76x76
38x38
19x19
Darknet53
x2
concat
x2
concat
Yolo Detection Block
Yolo Detection Block
Yolo Detection Block
Output conv
Output conv
Output conv
19x19
38x38
76x76

SSD: Single Shot Detector

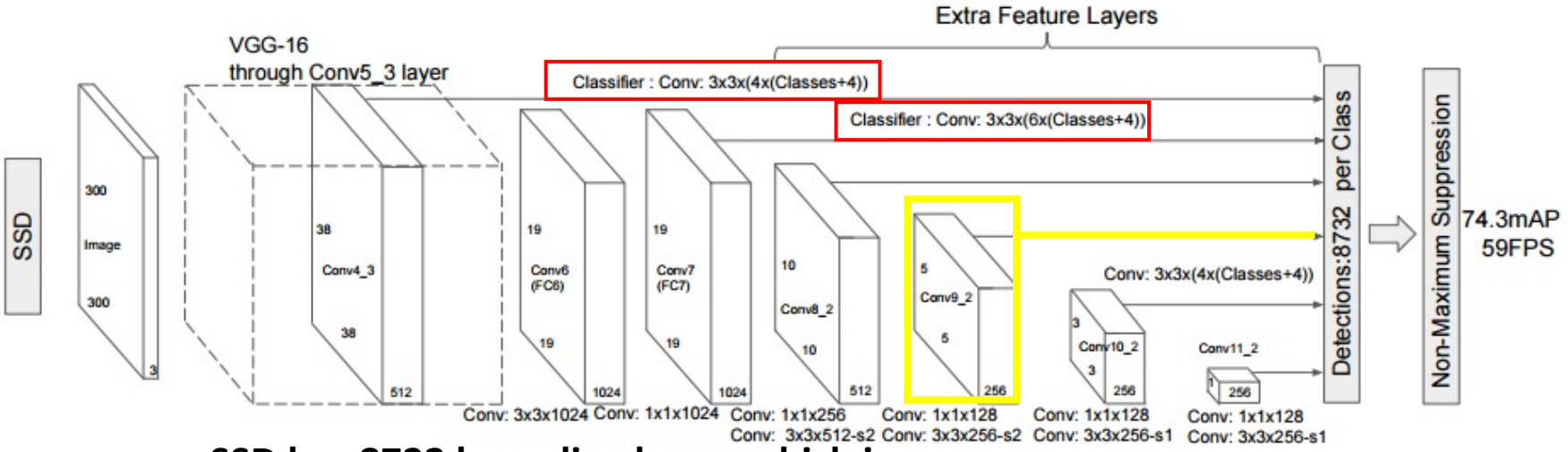
- Similar to YOLO but the box grid is denser, there are many grids with different box sizes
- The backbone network architecture is different from YOLO
- Data augmentation + Hard negative mining



(a) Image with GT boxes (b) 8×8 feature map (c) 4×4 feature map

SSD: Single Shot Detector

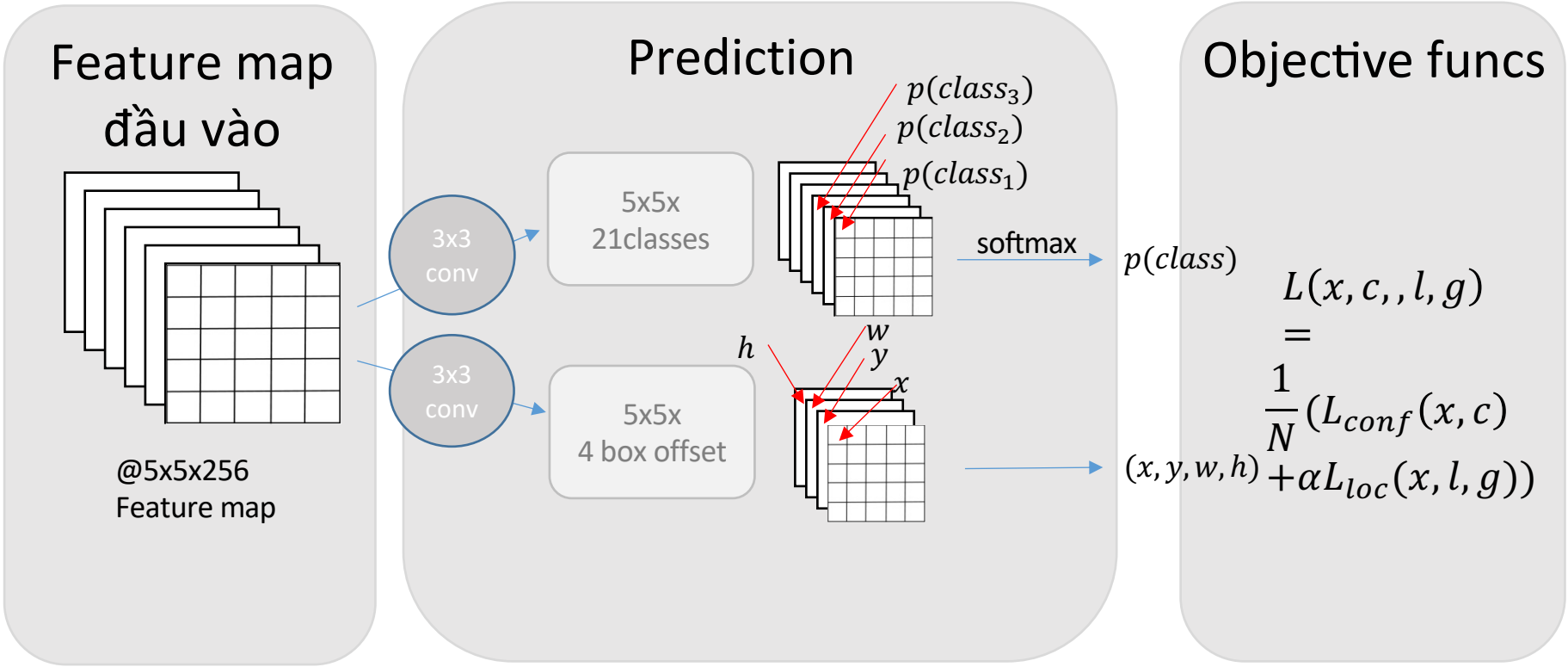
- Network backbone: VGG-16
- Add additional convolutional layers behind the layers of the backbone network
- Detect objects at different levels in the network (Multi-scale)

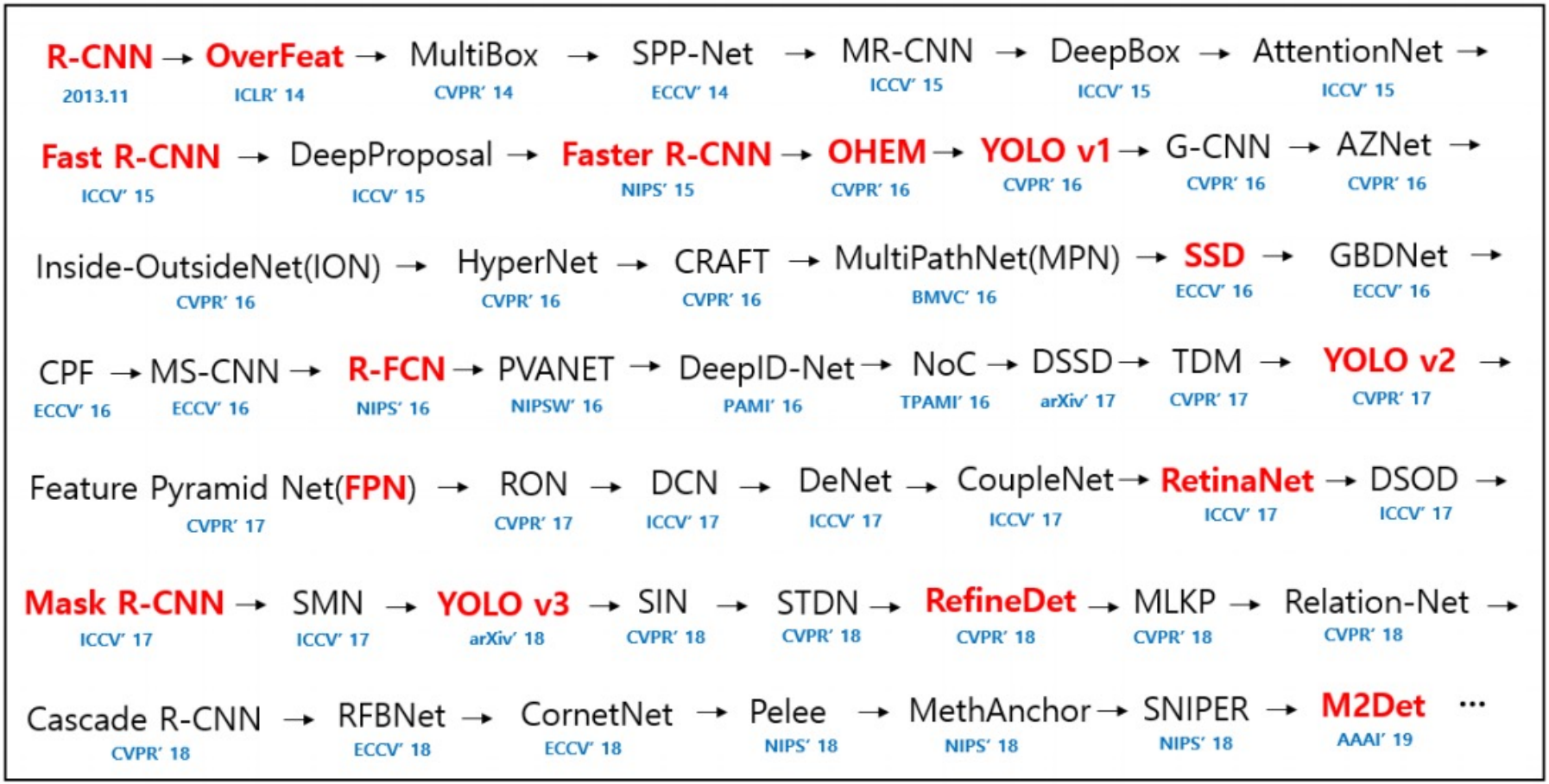


SSD has 8732 bounding boxes which is more than that of YOLO.

Liu et al. ECCV 2016.

SSD: Single Shot Detector




R-CNN 2013.11
OverFeat ICLR' 14
MultiBox CVPR' 14
SPP-Net ECCV' 14
MR-CNN ICCV' 15
DeepBox ICCV' 15
AttentionNet ICCV' 15
Fast R-CNN ICCV' 15
DeepProposal ICCV' 15
Faster R-CNN NIPS' 15
OHEM CVPR' 16
YOLO v1 CVPR' 16
G-CNN CVPR' 16
AZNet CVPR' 16
Inside-OutsideNet(ION) CVPR' 16
HyperNet CVPR' 16
CRAFT CVPR' 16
MultiPathNet(MPN) BMVC' 16
SSD ECCV' 16
GBDNet ECCV' 16
CPF ECCV' 16
MS-CNN ECCV' 16
R-FCN NIPS' 16
PVANET NIPSW' 16
DeepID-Net PAMI' 16
NoC TPAMI' 16
DSSD arXiv' 17
TDM CVPR' 17
YOLO v2 CVPR' 17
Feature Pyramid Net(FPN) CVPR' 17
RON CVPR' 17
DCN ICCV' 17
DeNet ICCV' 17
CoupleNet ICCV' 17
RetinaNet ICCV' 17
DSOD ICCV' 17
Mask R-CNN ICCV' 17
SMN ICCV' 17
YOLO v3 arXiv' 18
SIN CVPR' 18
STDN CVPR' 18
RefineDet CVPR' 18
MLKP CVPR' 18
Relation-Net CVPR' 18
Cascade R-CNN CVPR' 18
RFBNet ECCV' 18
CornetNet ECCV' 18
Pelee NIPS' 18
MethAnchor NIPS' 18
SNIPER NIPS' 18
M2Det AAAI' 19
...

One-stage vs two-stage

Faster and simpler

one-stage object detector

(dense sampling of object locations, scales, and aspect ratios)

YOLO

YOLO-v2

YOLO-v3

SSD

DSSD

MDCN

SqueezeNet

RetinaNet

RedefineDet

CornetNet

CenterNet

EfficientDet

More accurate

two-stage object detector

(proposal-driven mechanism)

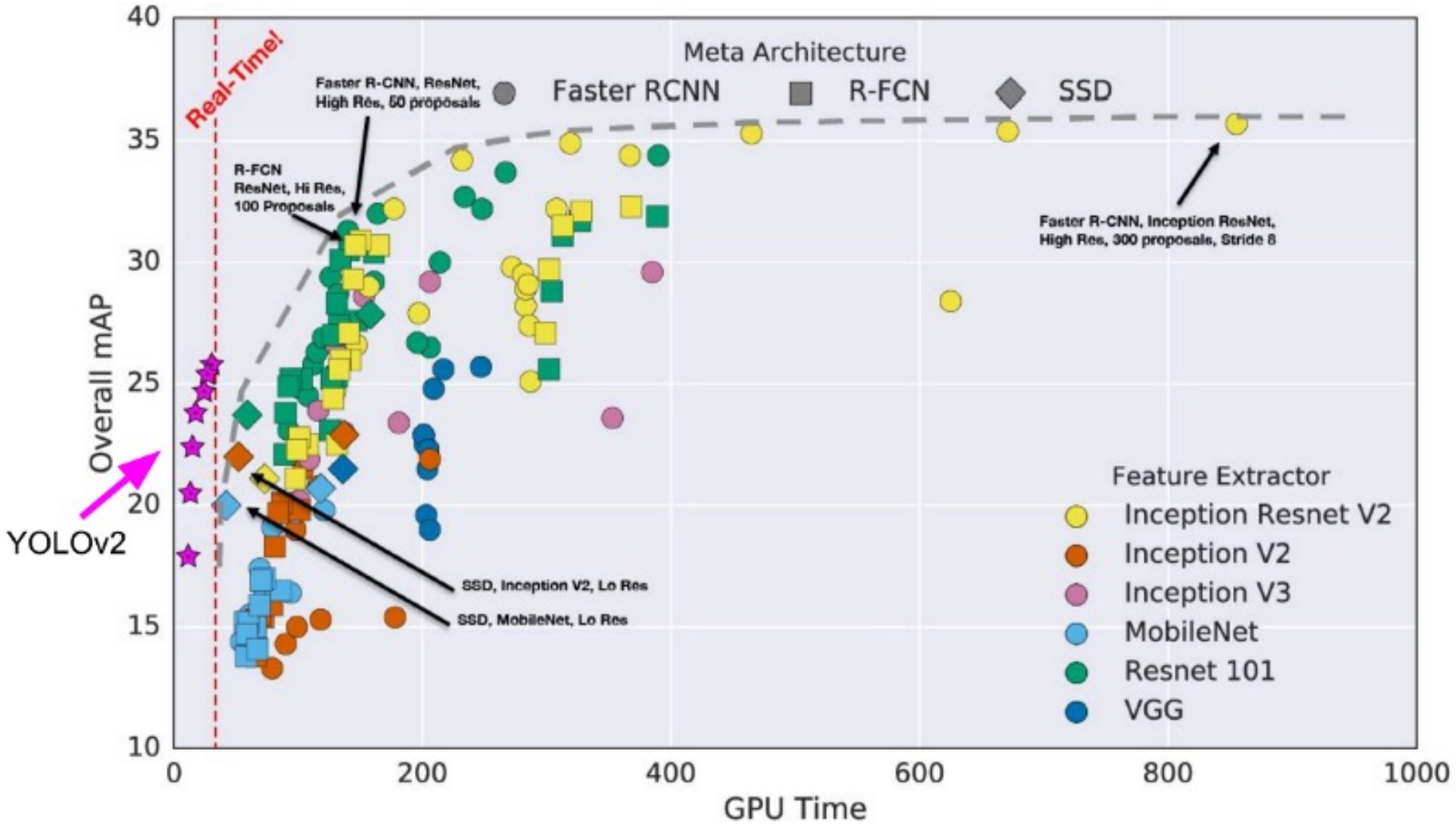
R-CNN

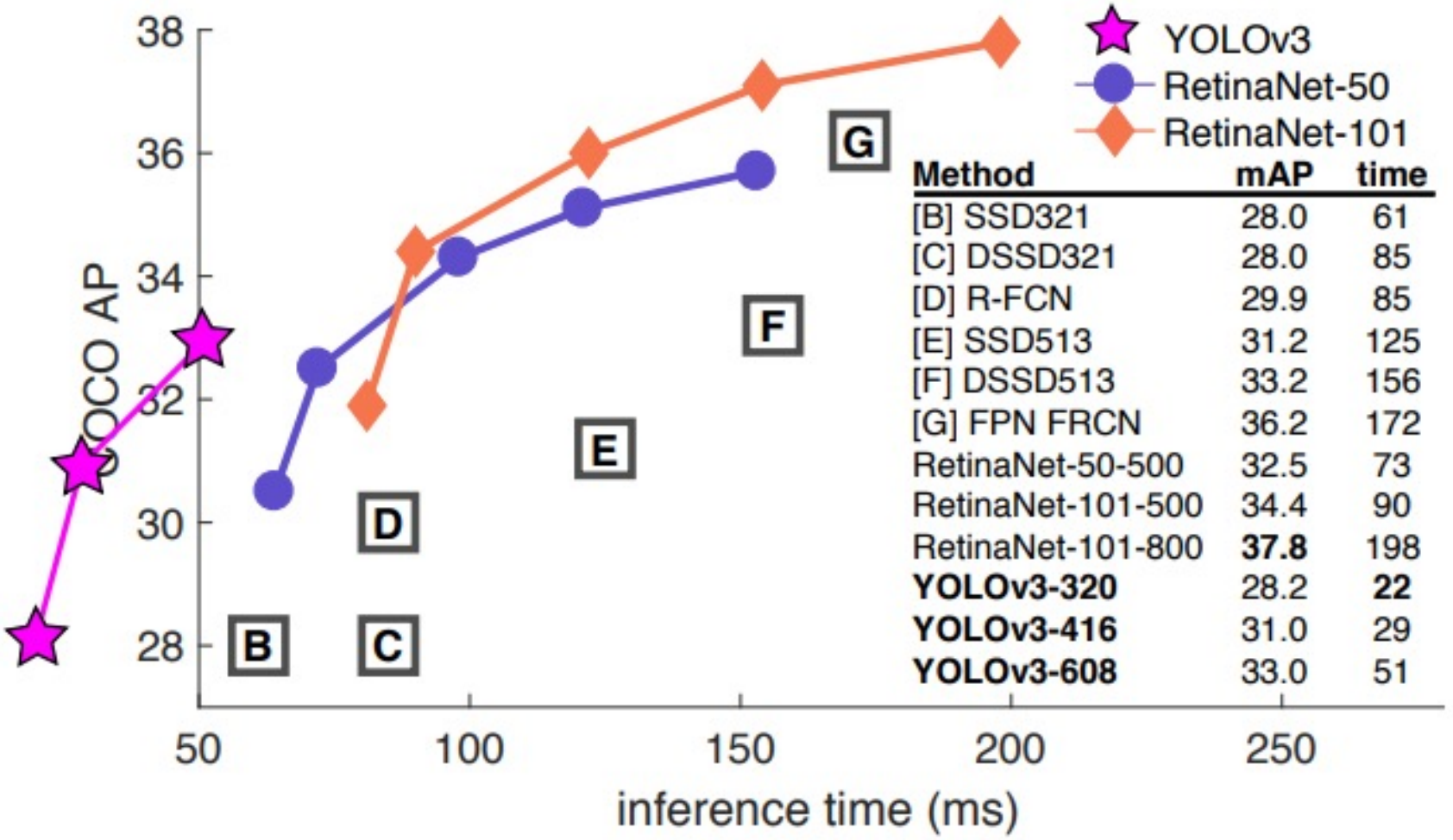
Fast R-CNN

Faster R-CNN

Feature Pyramid Network (FPN)

Mask R-CNN


Real-Time!
Meta Architecture
Faster RCNN
R-FCN
SSD
Faster R-CNN, ResNet, High Res, 50 proposals
R-FCN ResNet, Hi Res, 100 Proposals
Faster R-CNN, Inception ResNet, High Res, 300 proposals, Stride 8
SSD, Inception V2, Lo Res
SSD, MobileNet, Lo Res
YOLOv2
Overall mAP
GPU Time
Feature Extractor
Inception Resnet V2
Inception V2
Inception V3
MobileNet
Resnet 101
VGG
40
35
30
25
20
15
10
0
200
400
600
800
1000



Method	mAP	time
[B] SSD321	28.0	61
[C] DSSD321	28.0	85
[D] R-FCN	29.9	85
[E] SSD513	31.2	125
[F] DSSD513	33.2	156
[G] FPN FRCN	36.2	172
RetinaNet-50-500	32.5	73
RetinaNet-101-500	34.4	90
RetinaNet-101-800	**37.8**	198
YOLOv3-320	28.2	**22**
YOLOv3-416	31.0	29
YOLOv3-608	33.0	51

ĐẠI HỌC BÁCH KHOA

25 YEARS ANNIVERSARY SOICT

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

Thank you for your attention!