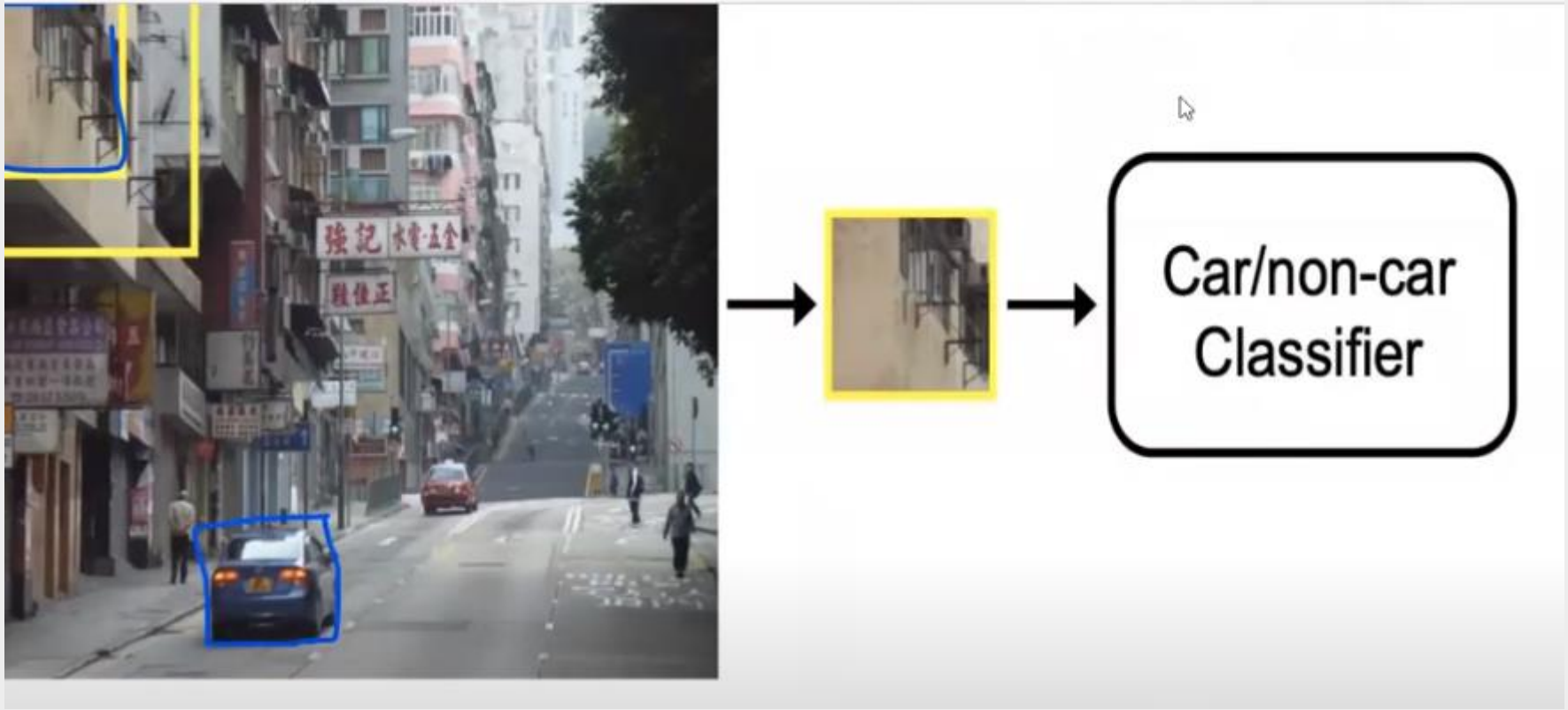

強記
Car/non-car
Classifier

Detection Datasets

VOC 2007 / 2012:

- 20 classes
- i.e. person, cat, dog, car, chair, bottle

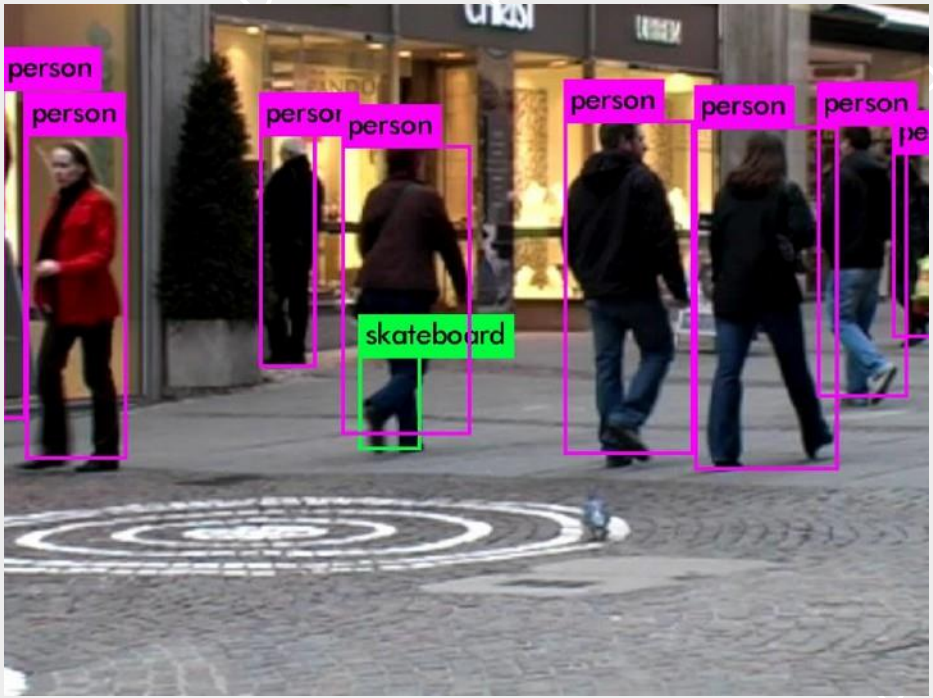
MS COCO:

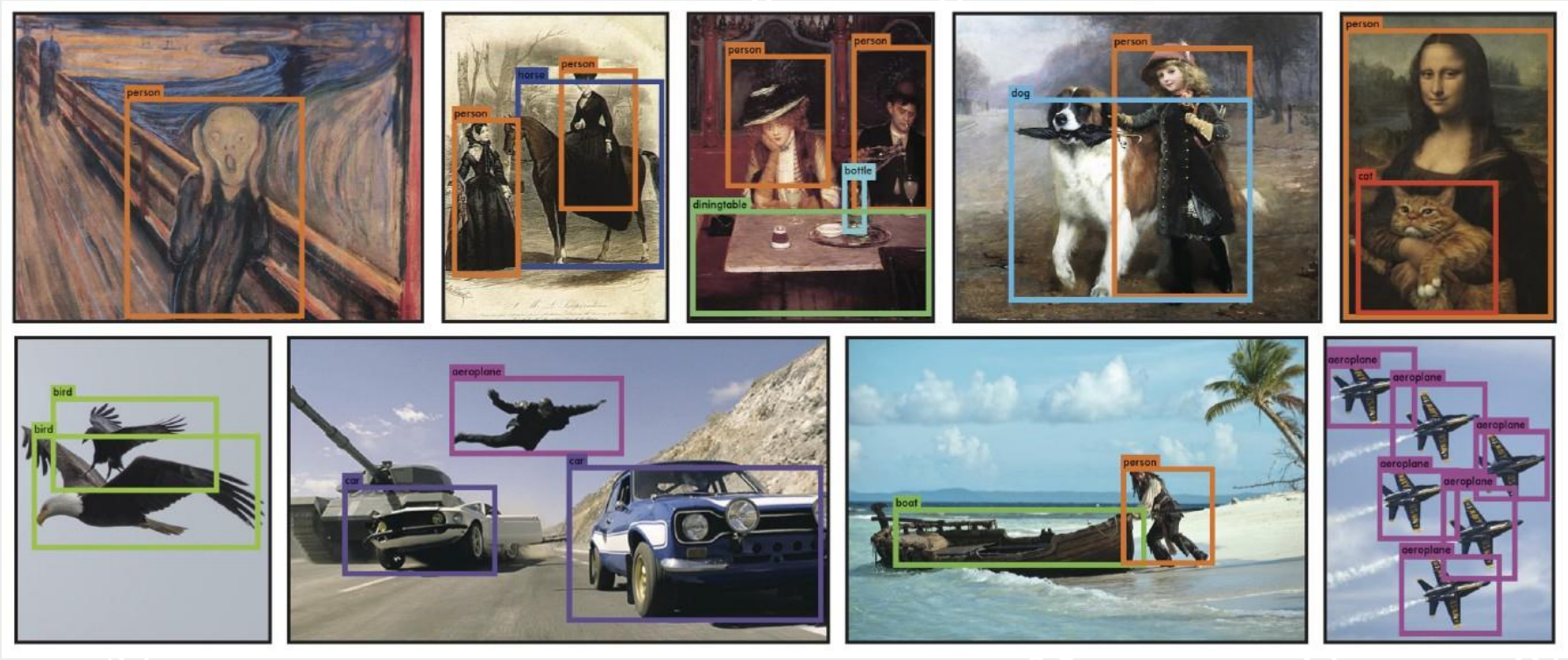
- 80 classes
- i.e. book, apple, teddy bear, scissors

ImageNet1000:

- 1000 classes
- i.e. German shepherd, golden retriever, European fire salamander

YOLO – The Target Output




person
person
horse
person
person
diningtable
person
bottle
person
dog
person
cat
bird
bird
aeroplane
car
car
person
boat
aeroplane
aeroplane
aeroplane
aeroplane
aeroplane
aeroplane

YOLO – The Detected Output

YOLO Versions

- YOLO (caffe) - https://github.com/xingwangsfu/caffe-yolo
- YOLO (tensorflow) - https://github.com/thtrieu/darkflow
- YOLO (darknet) - https://pjreddie.com/darknet/yolov1/ (C++)
- YOLO v2 (darknet) - https://pjreddie.com/darknet/yolov2/

  (C++) Better and faster - 91 fps for 288 x 288
- YOLO v3 (darknet) - https://pjreddie.com/darknet/yolo/ (C++)

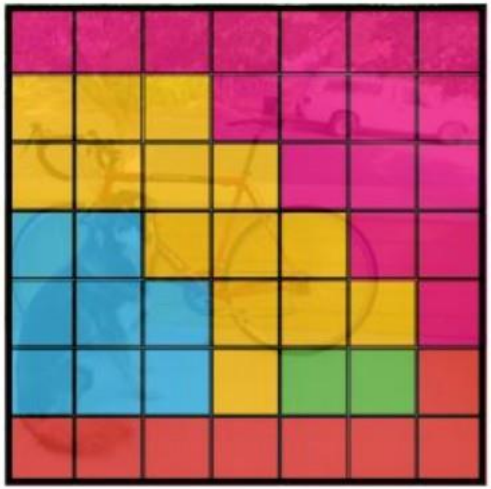
YOLO as Object Detection and Localization

Detection Frameworks	Train	mAP	FPS
Fast R-CNN [5]	2007+2012	70.0	0.5
Faster R-CNN VGG-16[15]	2007+2012	73.2	7
Faster R-CNN ResNet[6]	2007+2012	76.4	5
YOLO [14]	2007+2012	63.4	45
SSD300 [11]	2007+2012	74.3	46
SSD500 [11]	2007+2012	76.8	19
YOLOv2 288 × 288	2007+2012	69.0	91
YOLOv2 352 × 352	2007+2012	73.7	81
YOLOv2 416 × 416	2007+2012	76.8	67
YOLOv2 480 × 480	2007+2012	77.8	59
YOLOv2 544 × 544	2007+2012	**78.6**	40

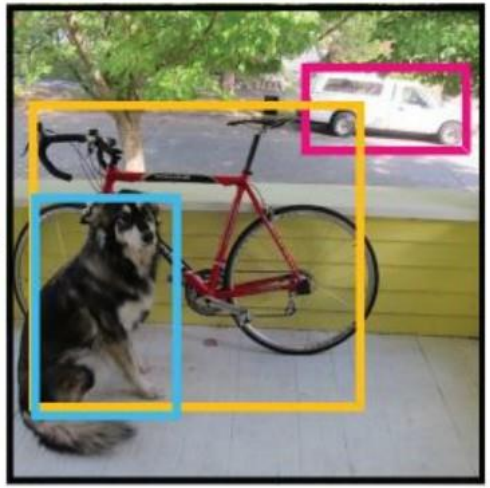
- PASCAL VOC 2007 as a dataset;
- different sizes of input for YOLOv2;
- all timing information is on a Geforce GTX Titan X.

mAP Mean Average Precision (Quality)
FPS Frames per second (processing)

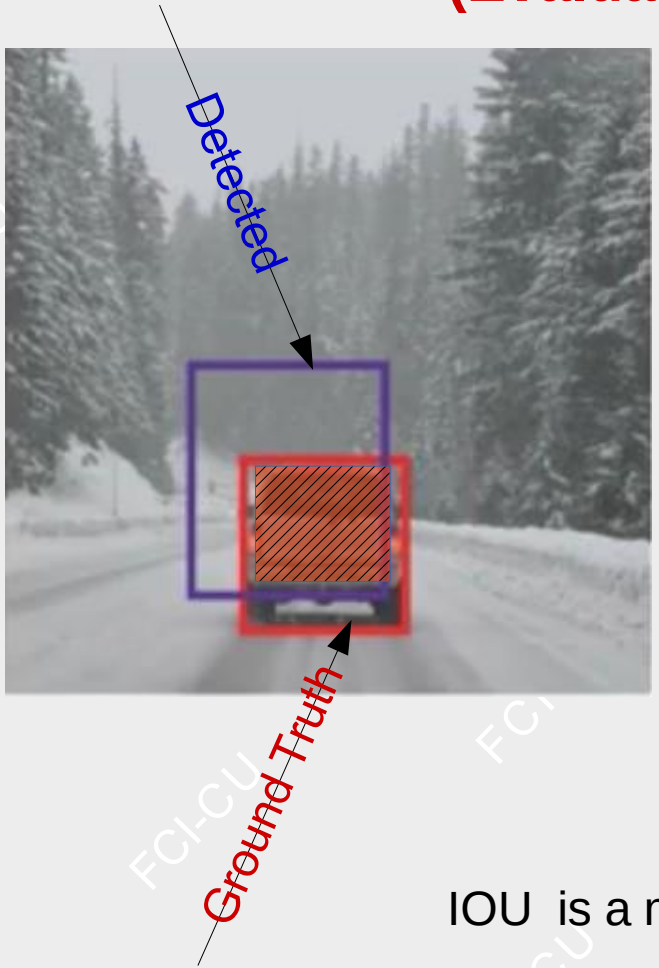
Detected Boxes Coordinates



Confidence of Detected Objects



Intersection Over Union (IOU)
(Evaluation of Detection Results Accuracy)



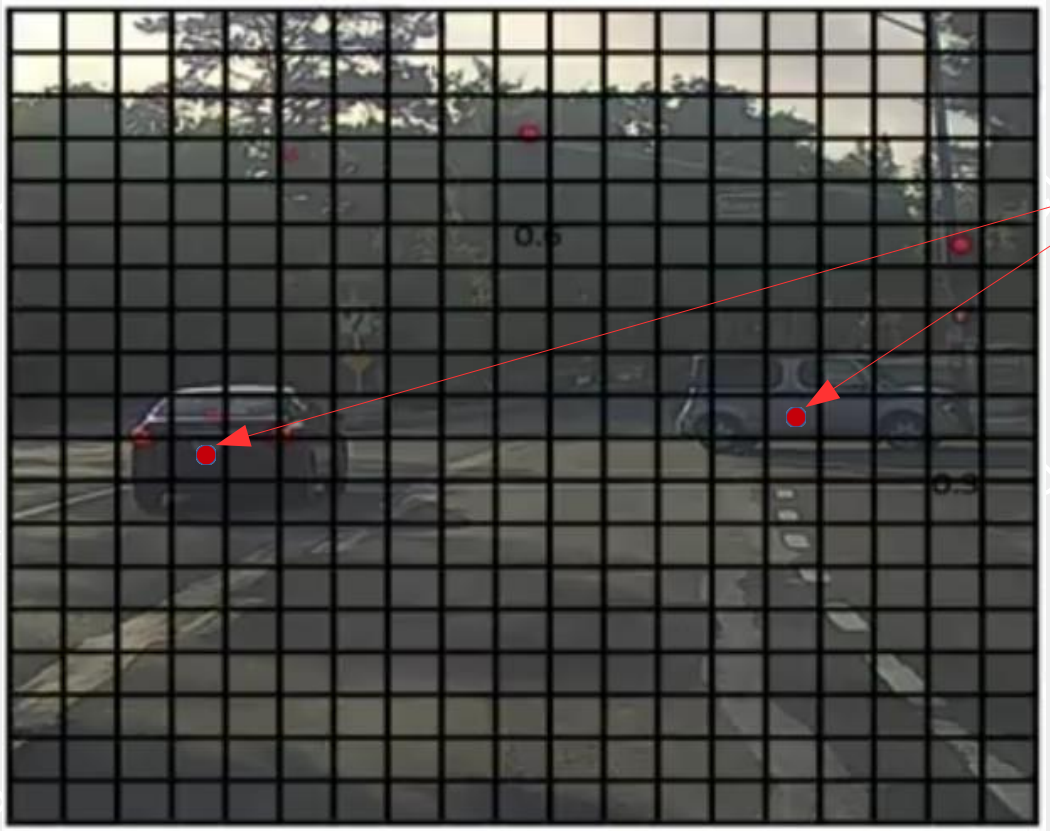
$$IOU = \frac{\text{Area of Union part}}{\text{Area of Intersection}}$$

Acceptance criteria IOU >0.5

(The higher the better)

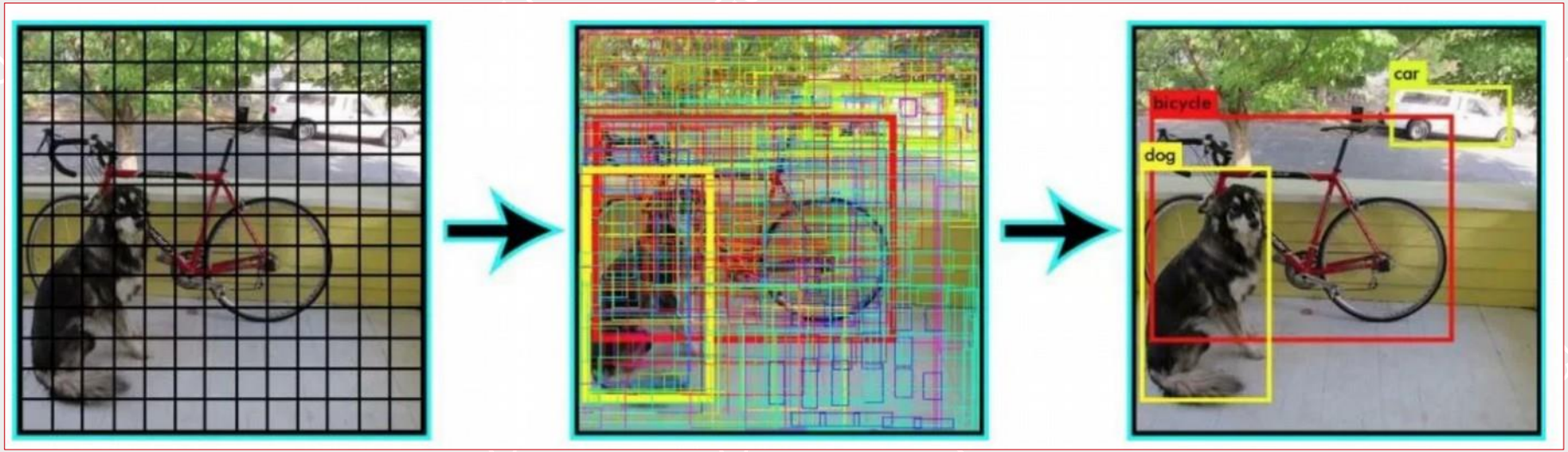
IOU is a measure of overlap between two bounding boxes

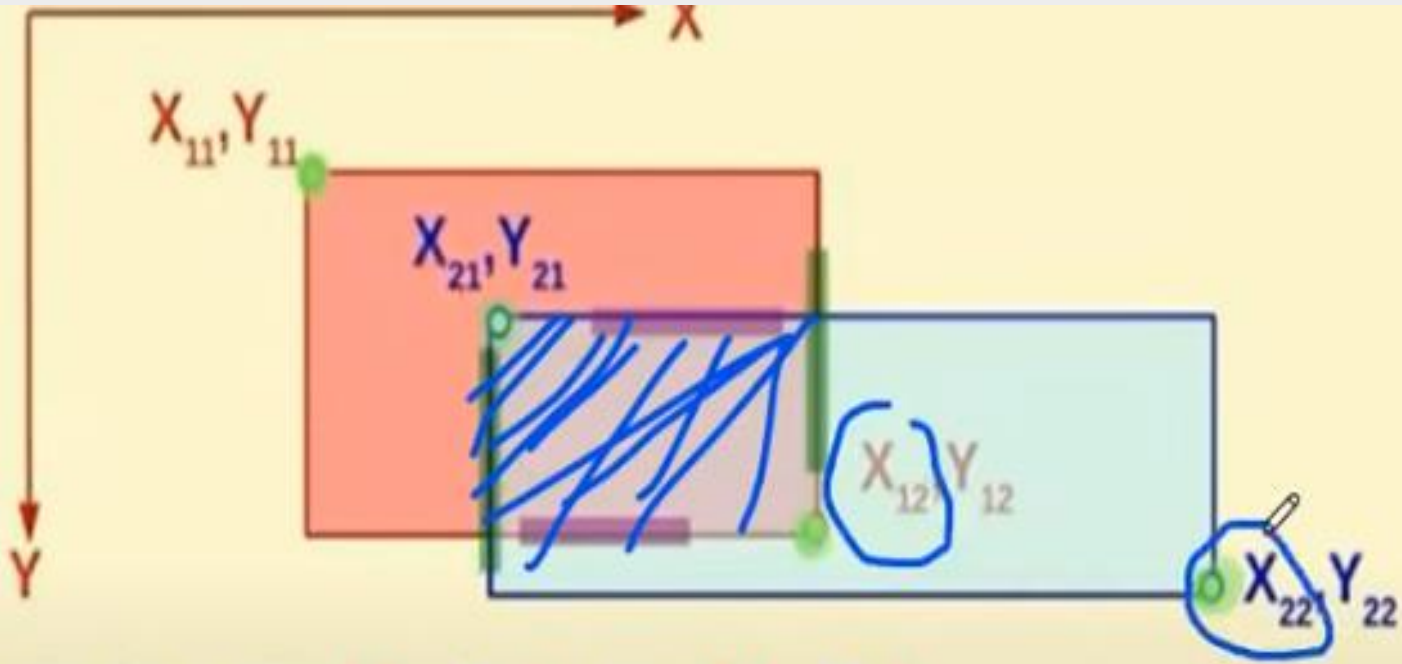
Non-Max Supression



Ground Truth
Objects centers and corresponding
Bounding Boxes

Remove Redundant Boxes using Non-Max Supression





Intersection Width = Min (X_{12}, X_{22}) - Max(X_{11}, X_{21})

Intersection Height = Min (Y_{12}, Y_{22}) - Max(Y_{11}, Y_{21})

Intersection AREA = Intersection Width * Intersection Width

Union AREA = ($X_{12} - X_{11}$) * ($Y_{12} - Y_{11}$) + ($X_{22} - X_{21}$) * ($Y_{22} - Y_{21}$) - Intersection AREA