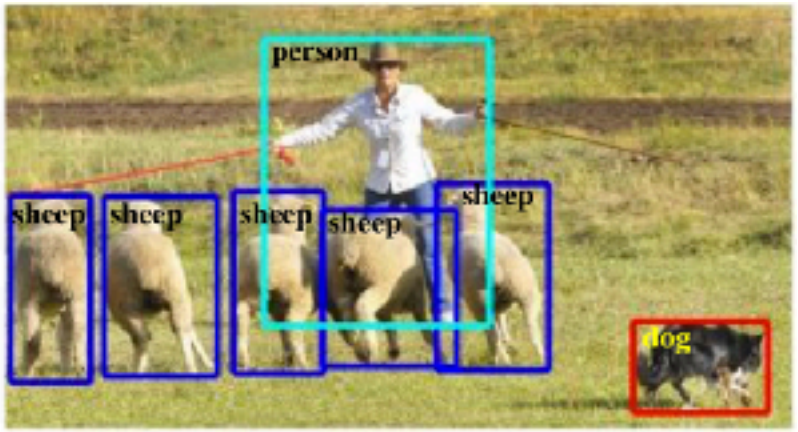
Recap: From Classification to Object Detection

(a) Object Classification



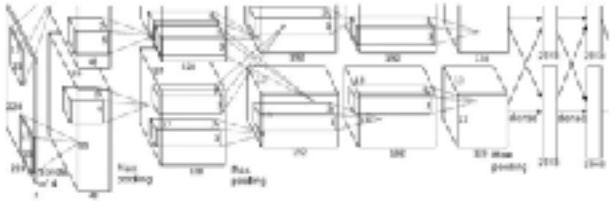
(b) Generic Object Detection
(Bounding Box)

Class

Class
(x,y,w,h)

Li Liu et al. Deep Learning for Generic Object Detection: A Survey 2018

Classification + Localization



Vector: 4096

Fully Connected: 4096 to 1000

Class Scores
Cat: 0.9
Dog: 0.05
Car: 0.01
...

Softmax Loss

Correct label: Cat

Fully Connected: 4096 to 4

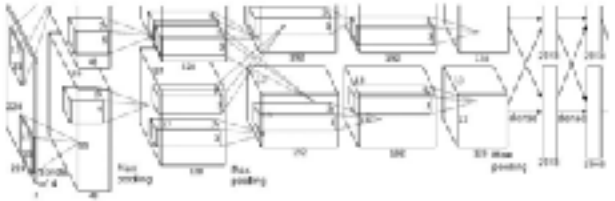
Box Coordinates (x, y, w, h)

L2 Loss

Correct box: (x', y', w', h')

Treat localization as a regression problem!

Classification + Localization



Correct label:
Cat

Fully Connected:
4096 to 1000

Class Scores
Cat: 0.9
Dog: 0.05
Car: 0.01
…

Softmax Loss

Multitask Loss

+

Loss

Vector:
4096

Fully Connected:
4096 to 4

Box Coordinates
(x, y, w, h)

L2 Loss

Correct box:
(x', y', w', h')

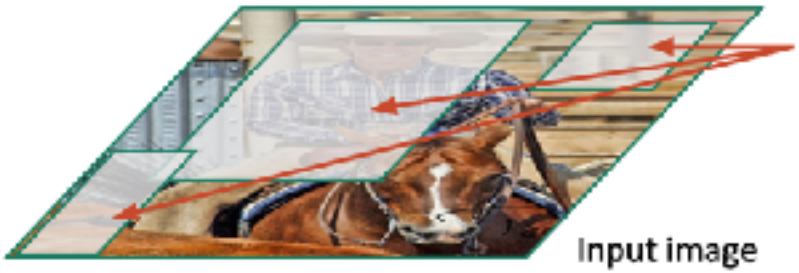
Treat localization as a regression problem!

Region CNN

Input image

Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

Region CNN

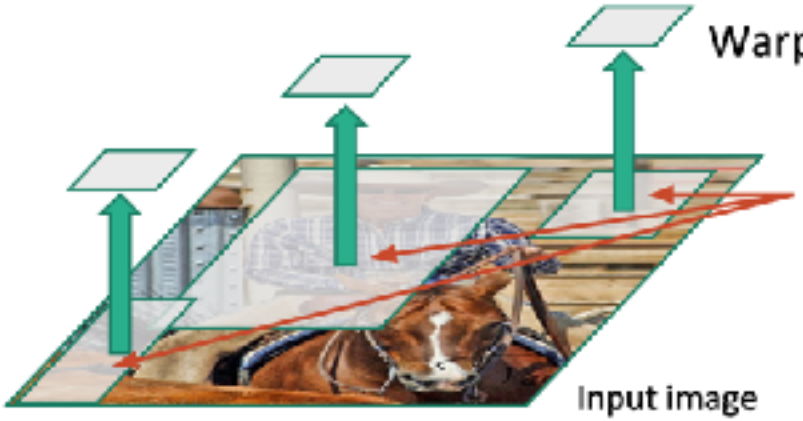


Input image

Regions of Interest (RoI) from a proposal method (~2k)

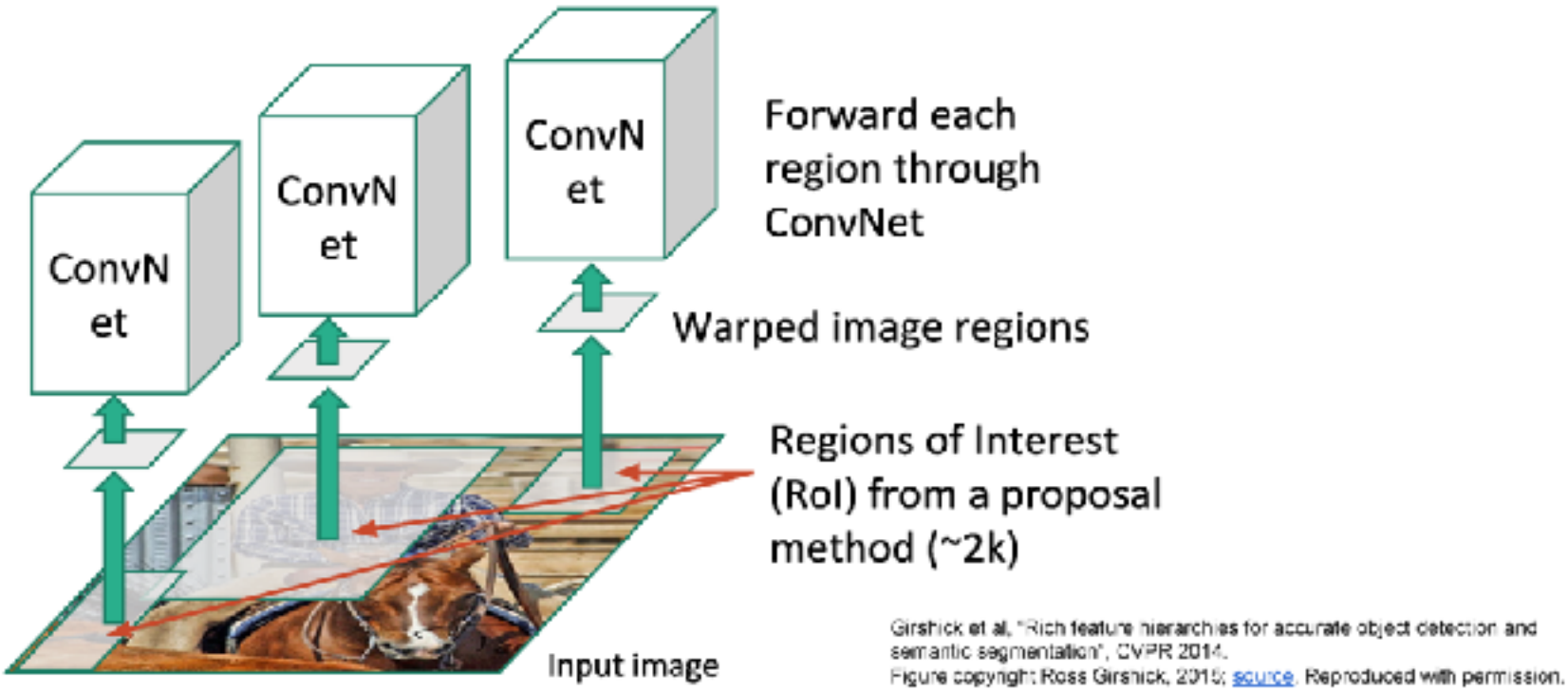
Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

Region CNN



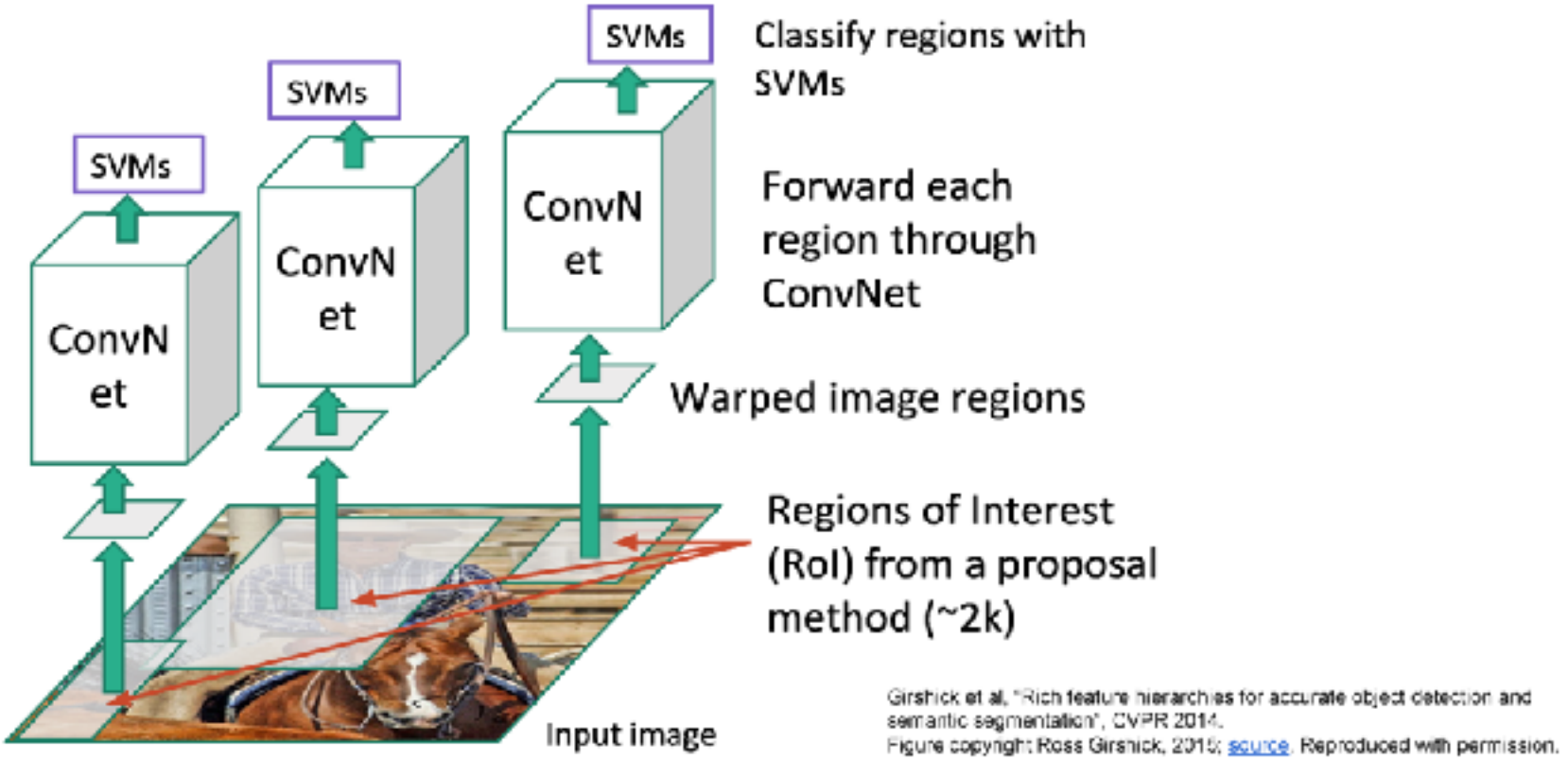
Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

Region CNN



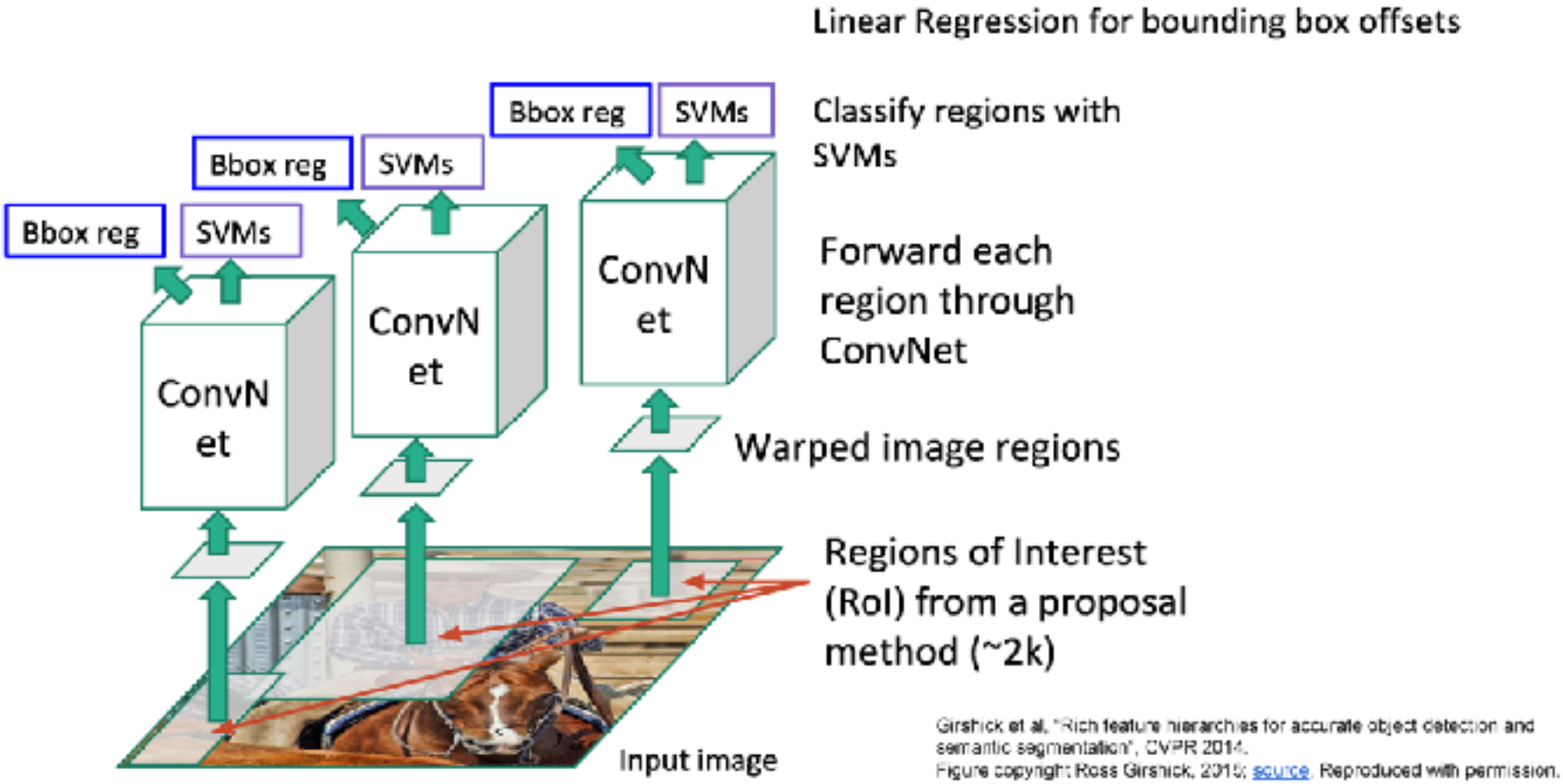
Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

Region CNN



Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

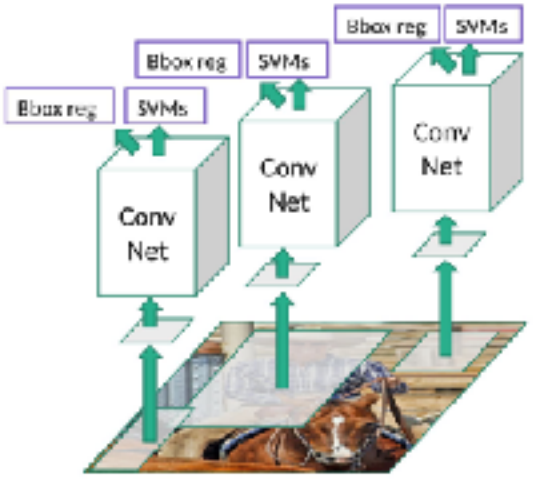
Region CNN



Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

Region 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]



Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
Slide copyright Ross Girshick, 2015; source. Reproduced with permission.

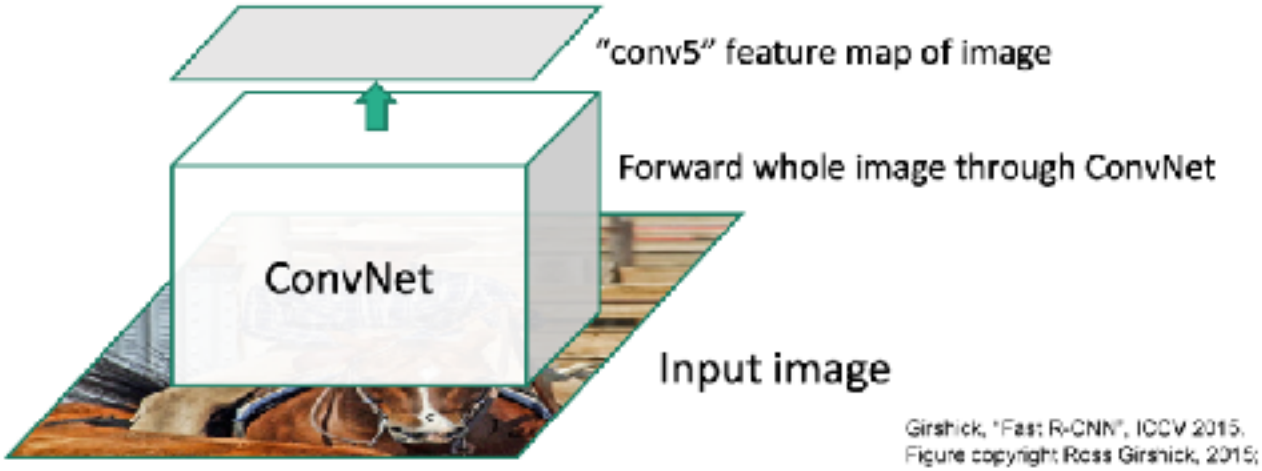
- Next: Fast RCNN, Faster RCNN

Fast RCNN

Input image

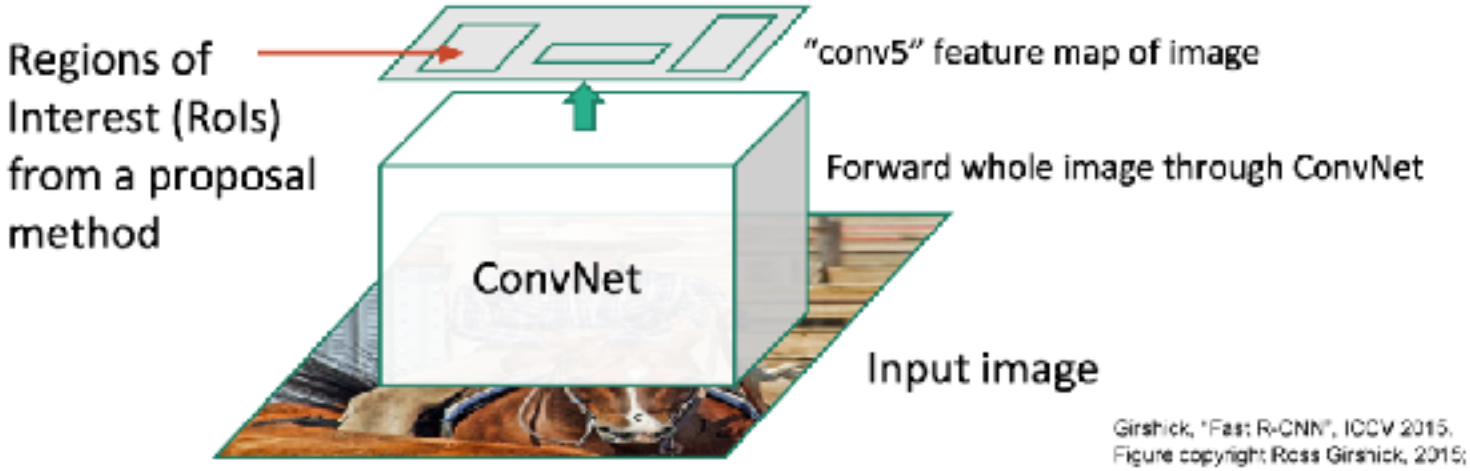
Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

Fast RCNN



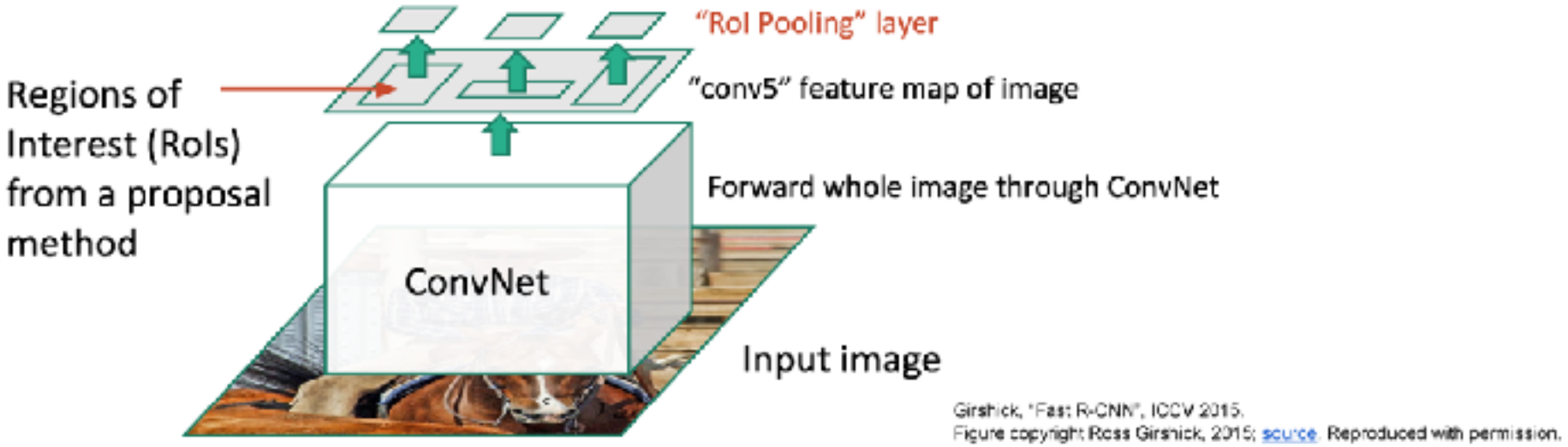
Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

Fast RCNN



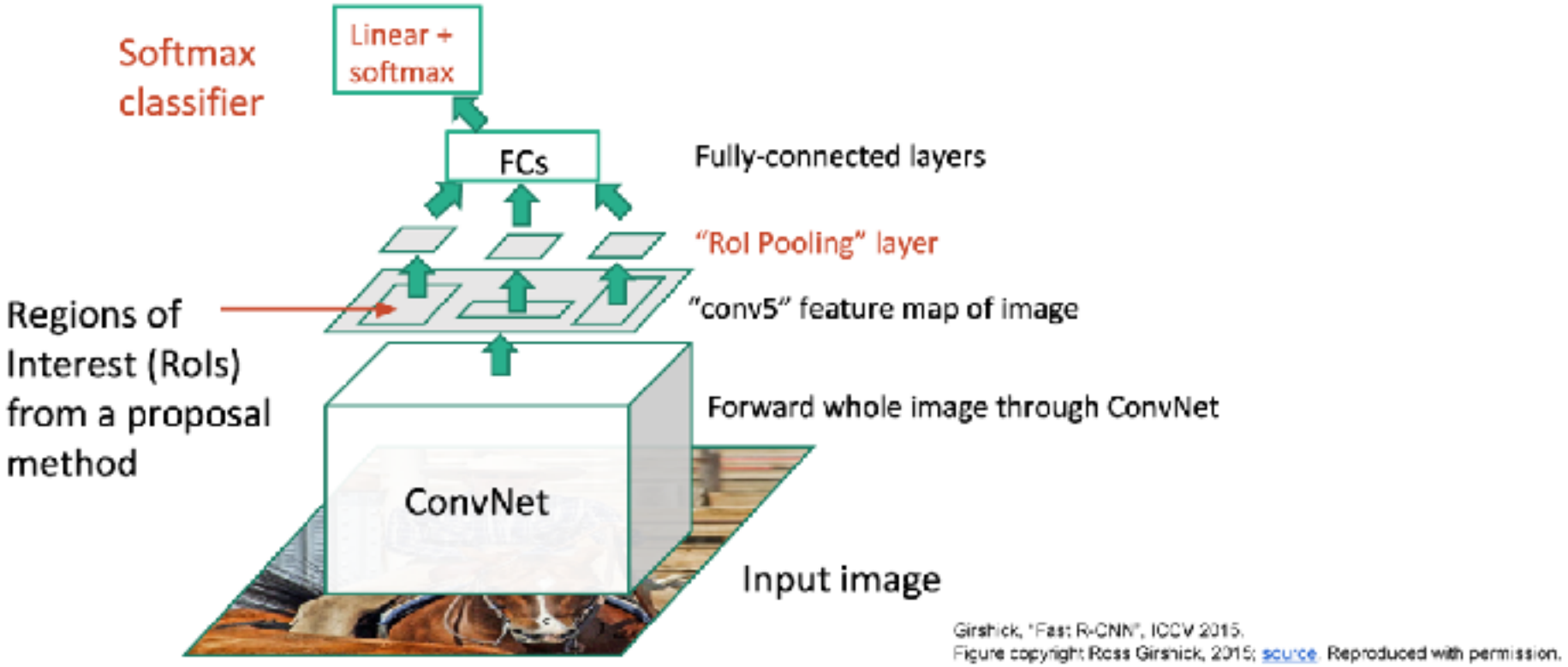
Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

Fast RCNN



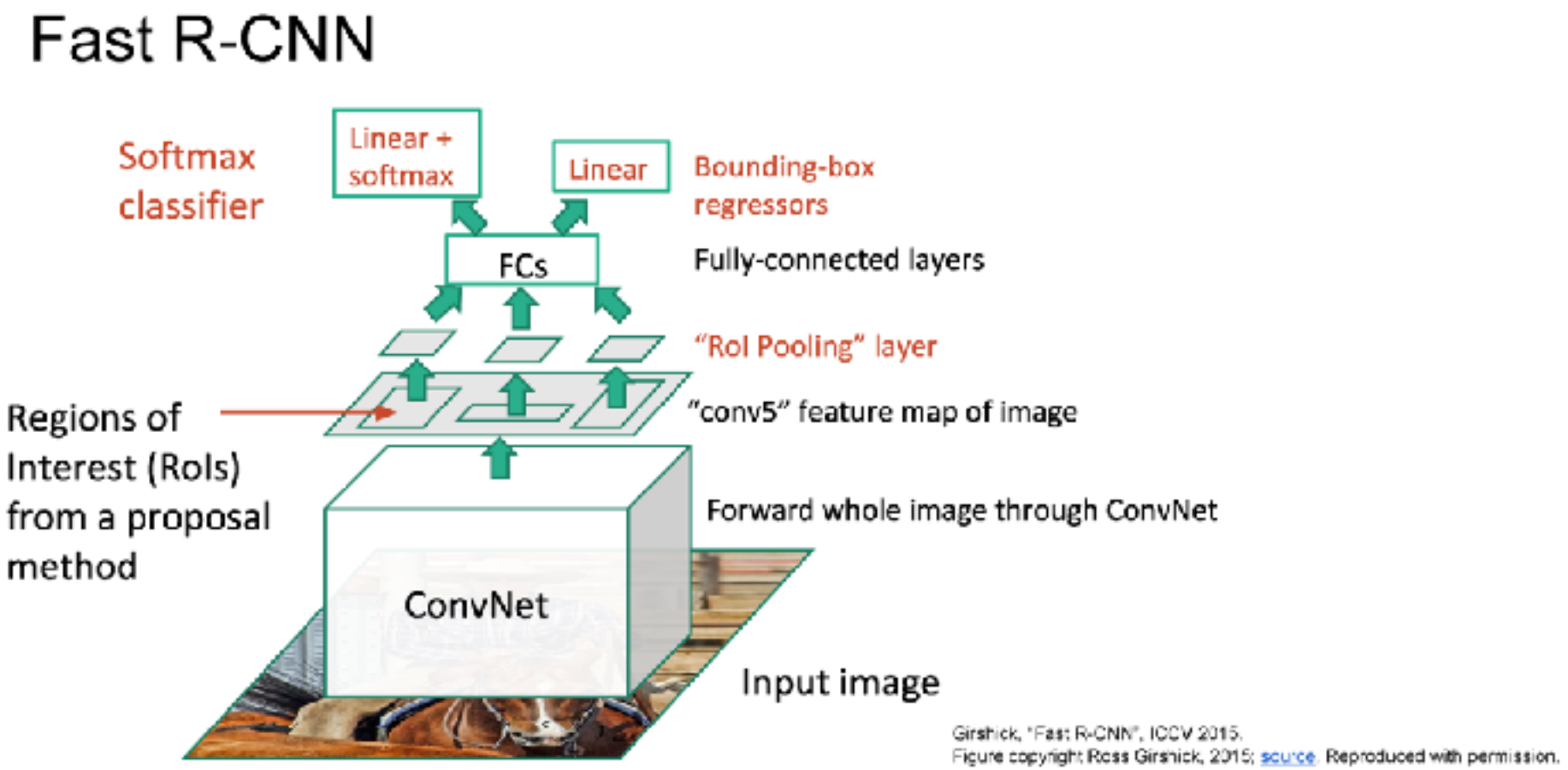
Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

Fast RCNN



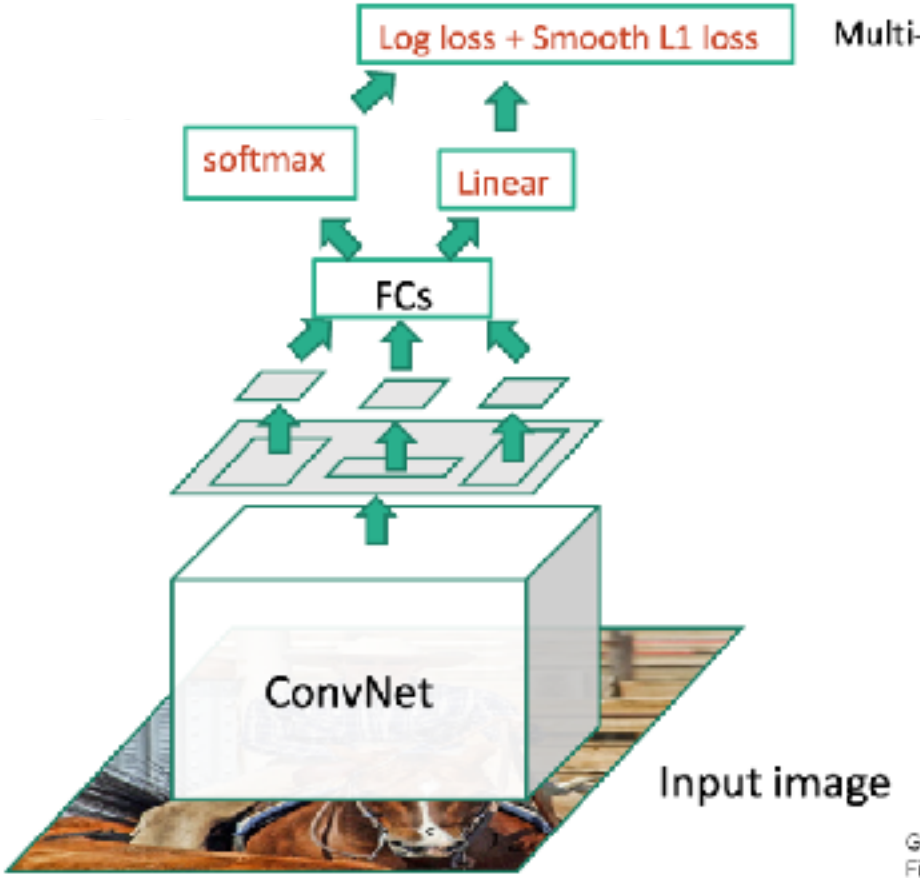
Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

Fast RCNN



Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

Fast RCNN: Training



Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

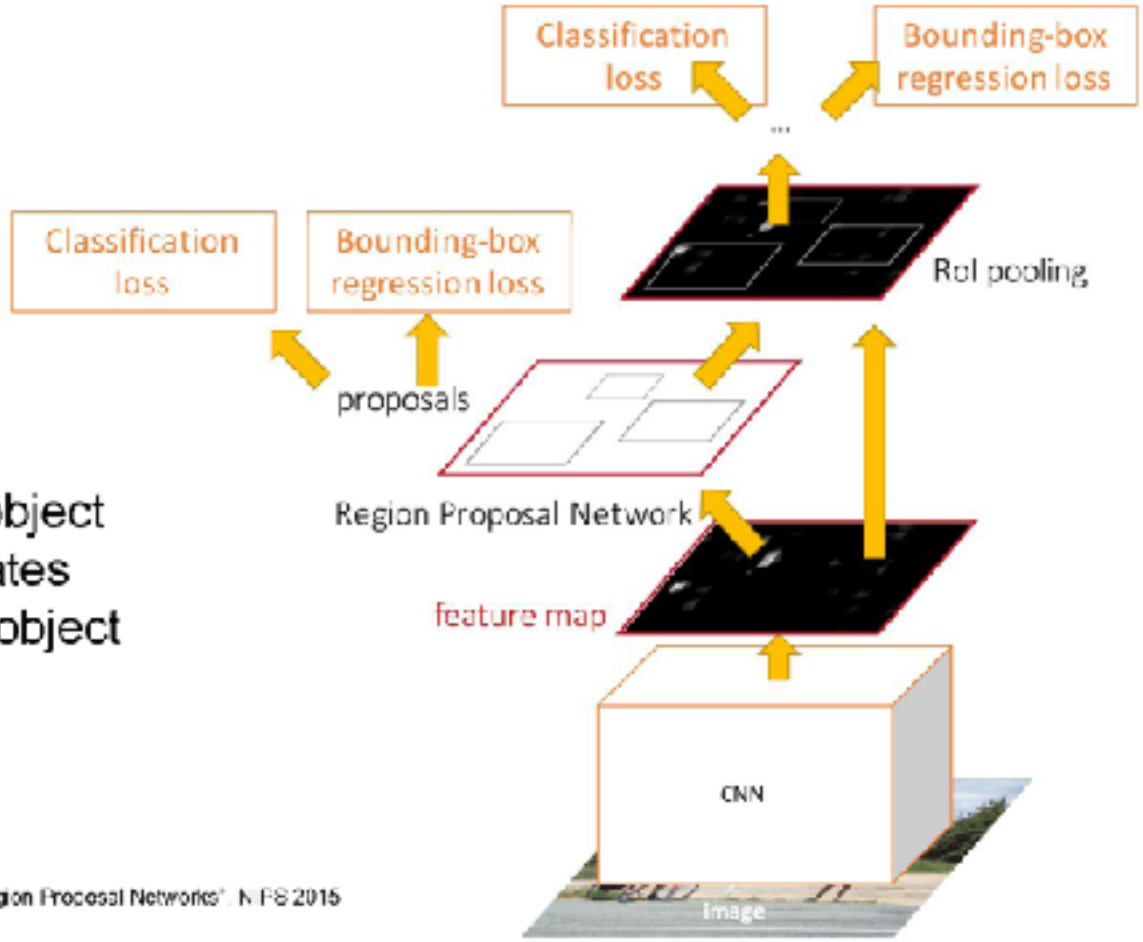
Faster RCNN

Make CNN do proposals!

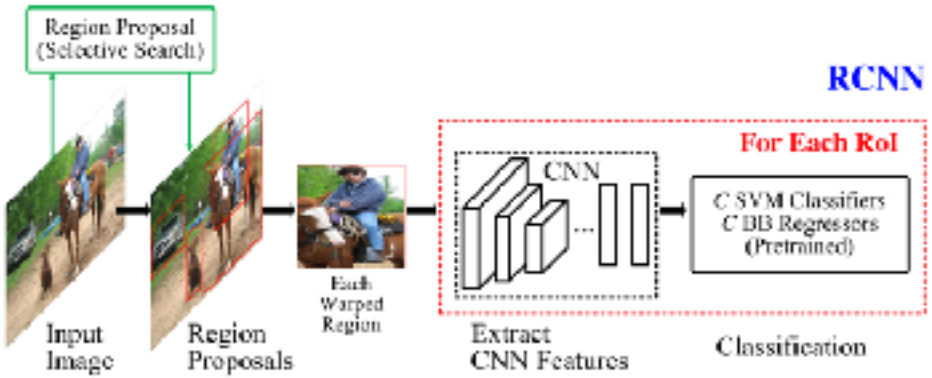
Insert **Region Proposal Network (RPN)** to predict proposals from features

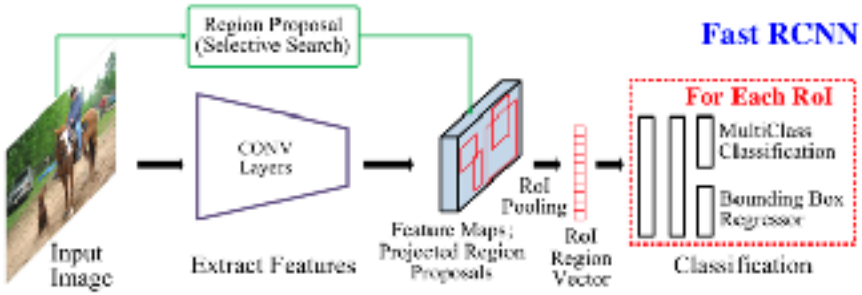
Jointly train with 4 losses:

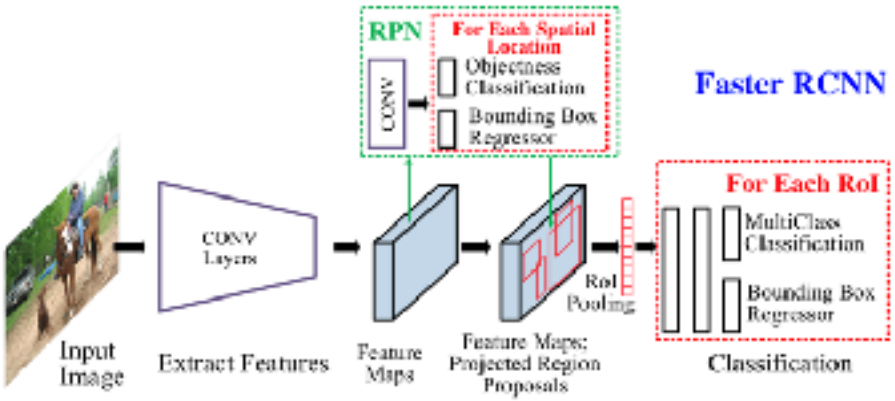
1. RPN classify object / not object
2. RPN regress box coordinates
3. Final classification score (object classes)
4. Final box coordinates



Ren et al, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NIPS 2015
Figure copyright 2015, Ross Girshick; reproduced with permission





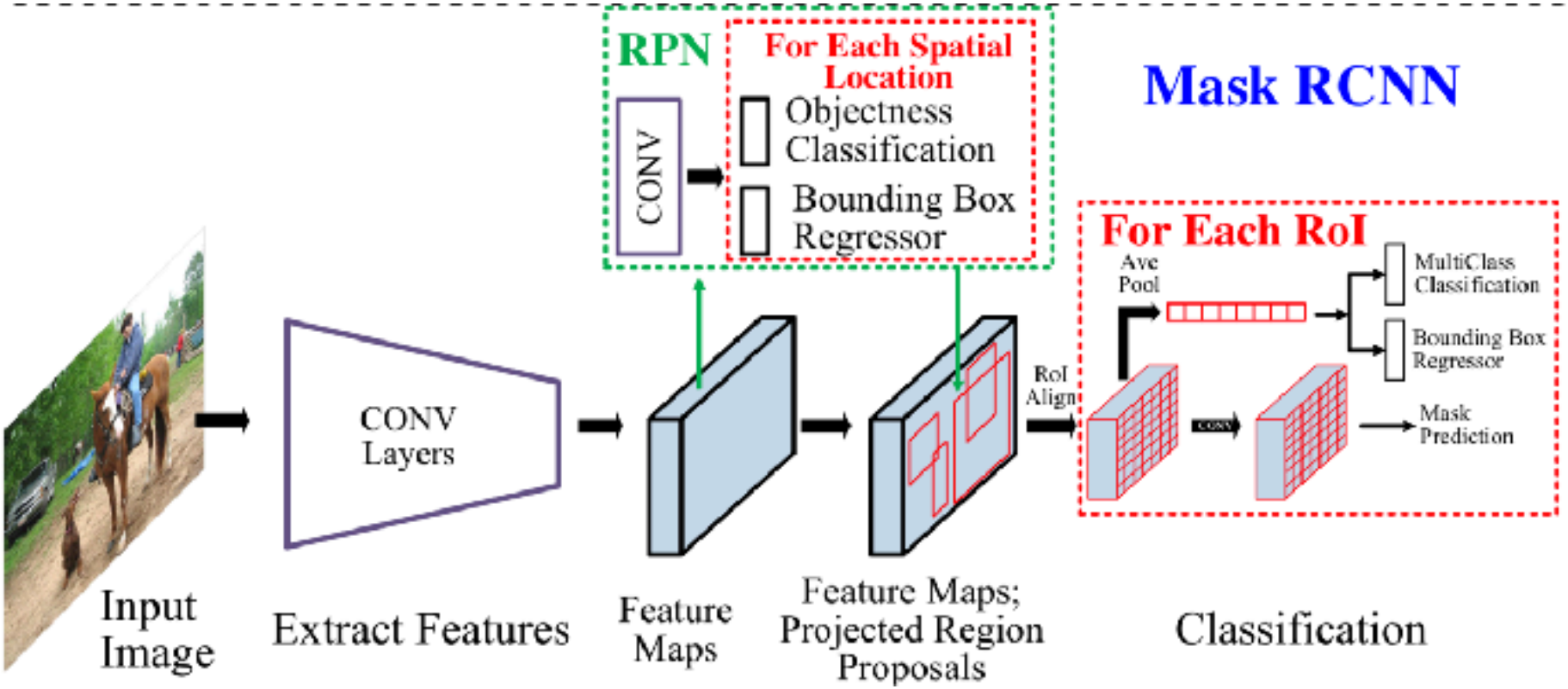


Li Liu et al. Deep Learning for Generic Object Detection: A Survey 2018

This Lecture

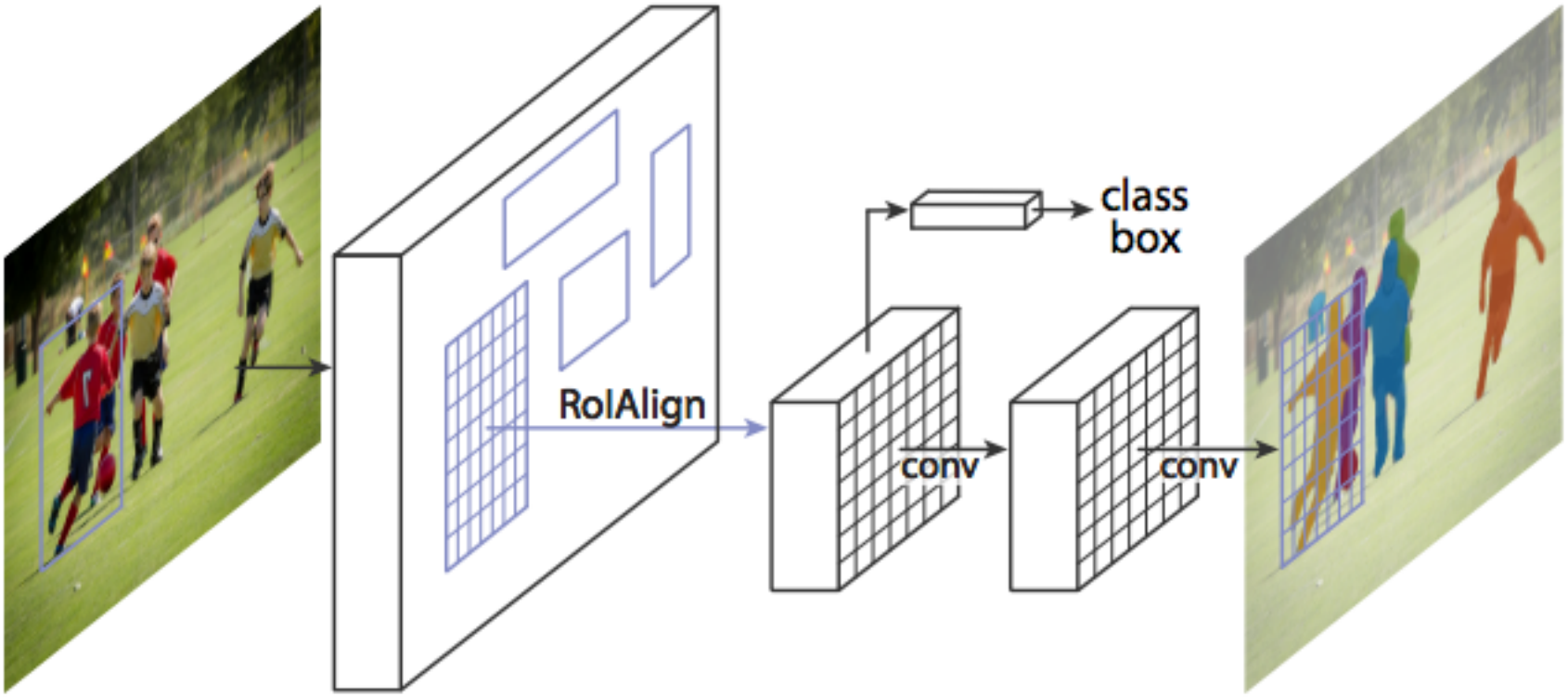
- Mask RCNN
- Yolo
- SSD
- 3D Object Detection

Mask RCNN



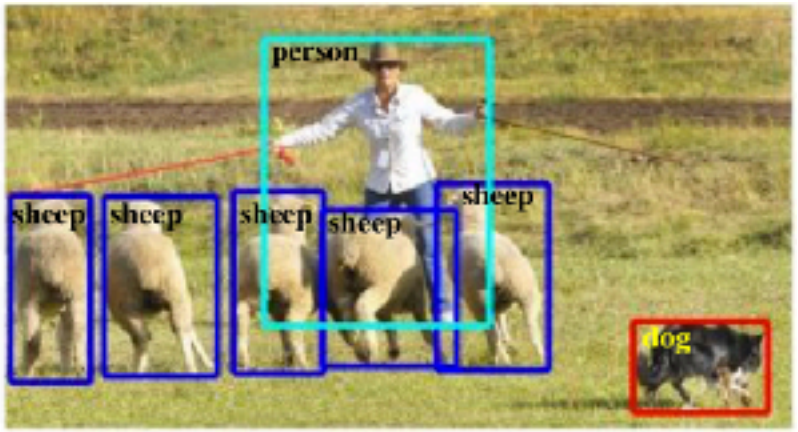
Li Liu et al. Deep Learning for Generic Object Detection: A Survey 2018

Mask RCNN: Mask Prediction

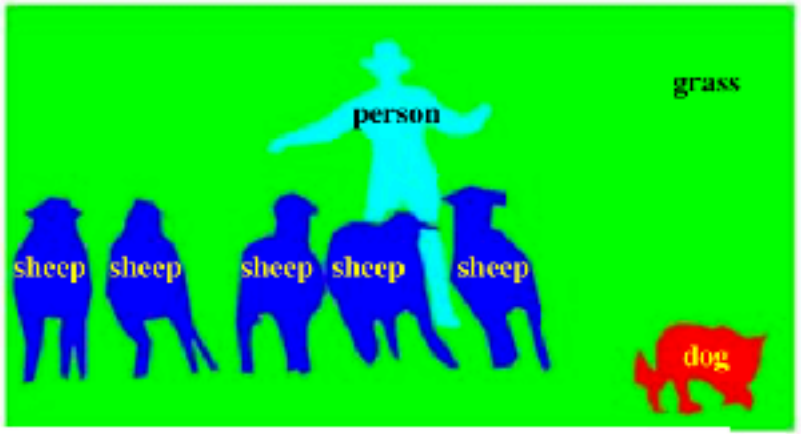


Mask RCNN

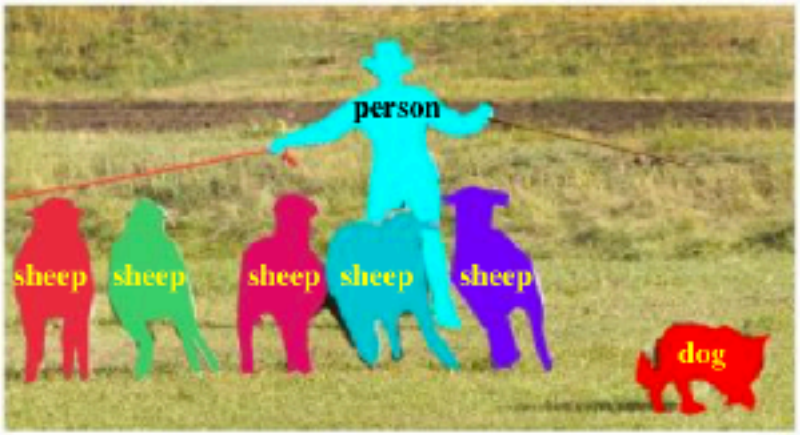
(a) Object Classification



(b) Generic Object Detection (Bounding Box)

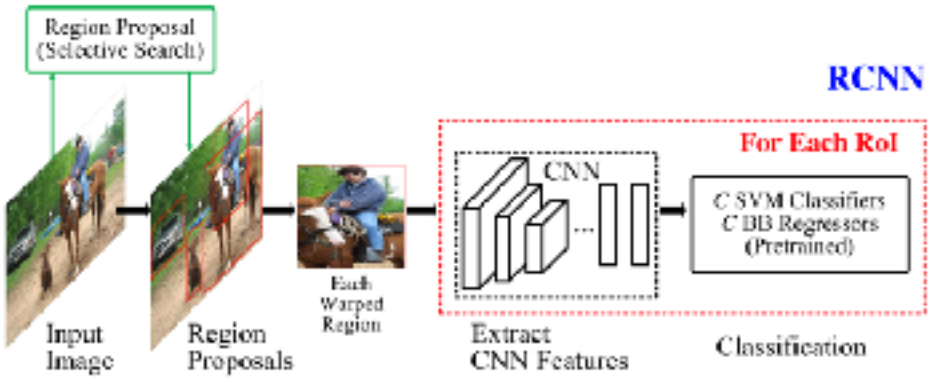


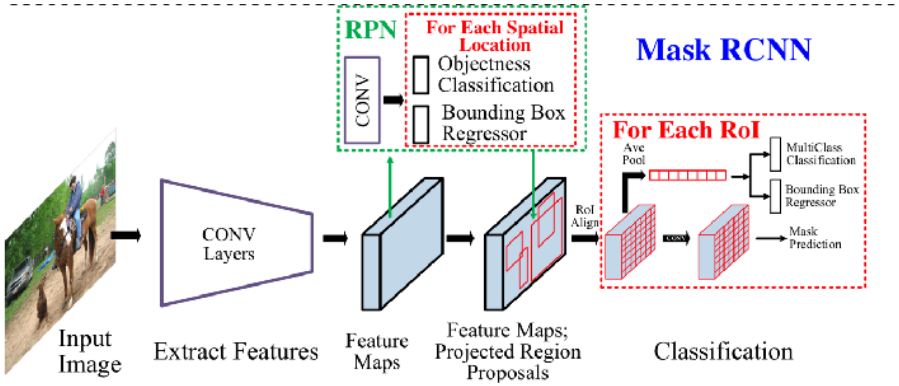
(c) Semantic Segmentation

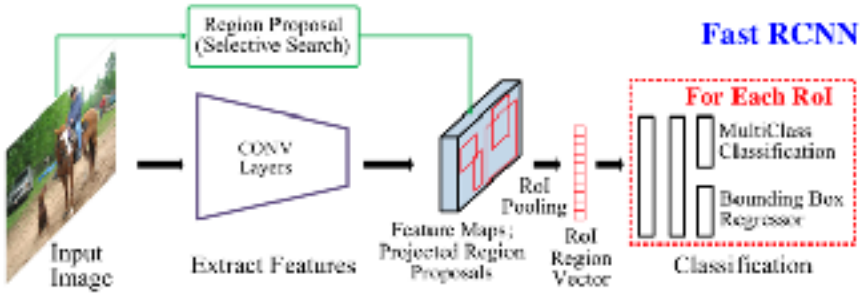


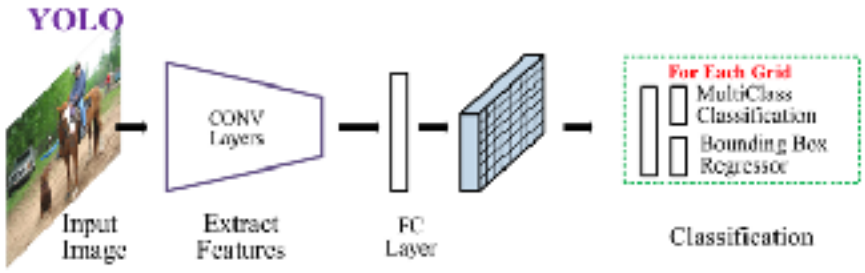
(d) Object Instance Segmetation

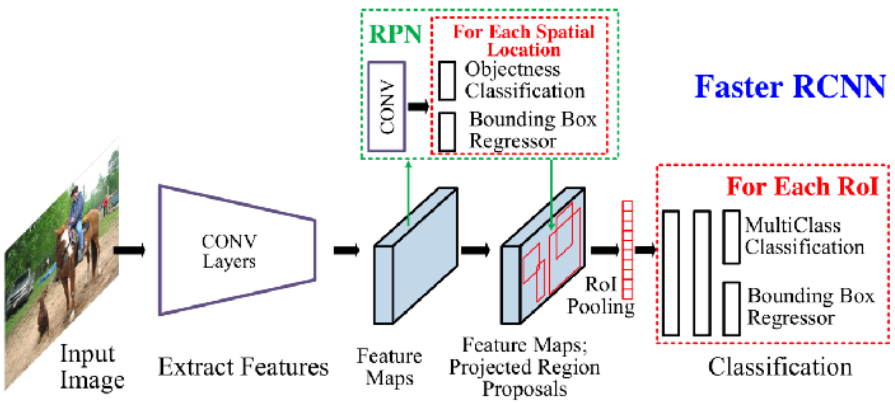
Li Liu et al. Deep Learning for Generic Object Detection: A Survey 2018

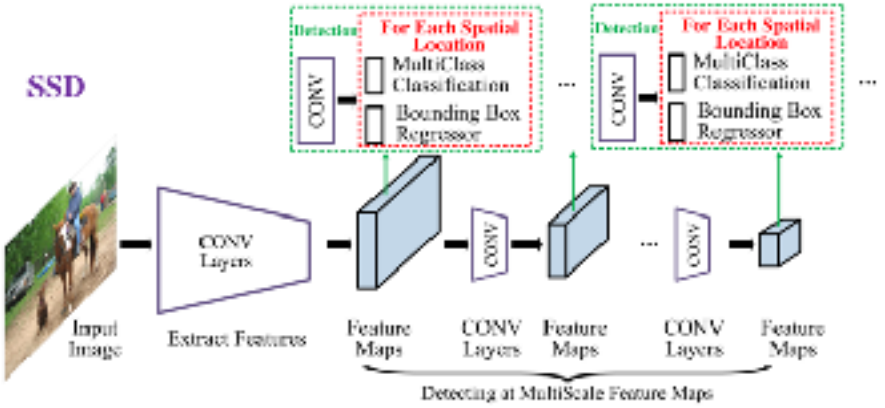






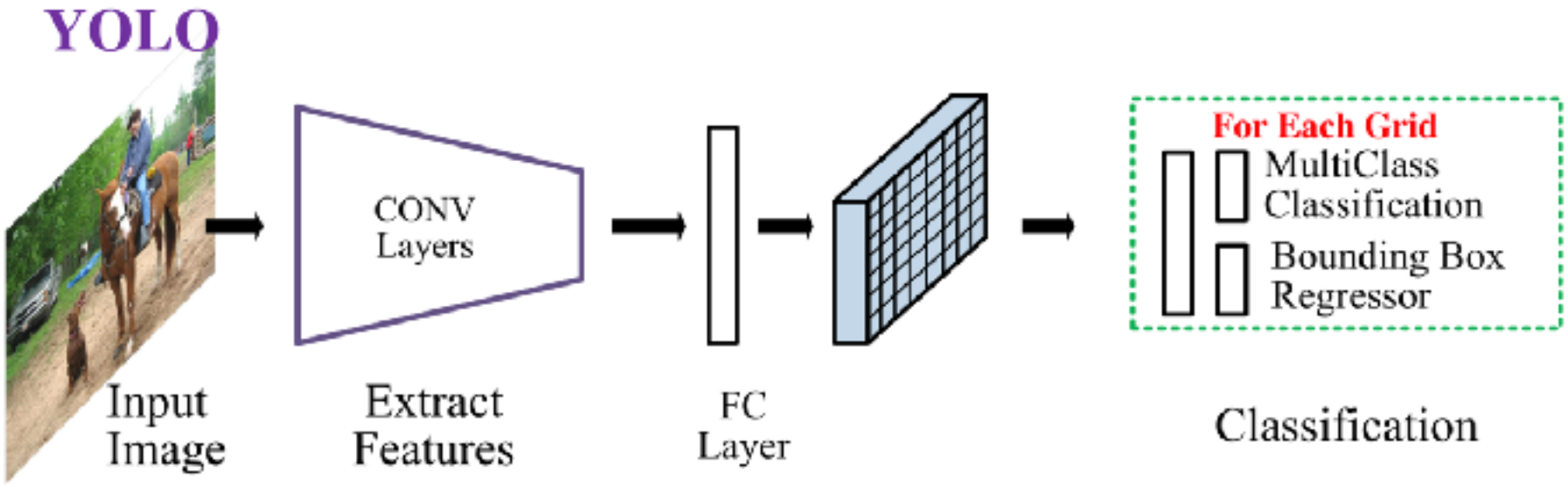






Li Liu et al. Deep Learning for Generic Object Detection: A Survey 2018

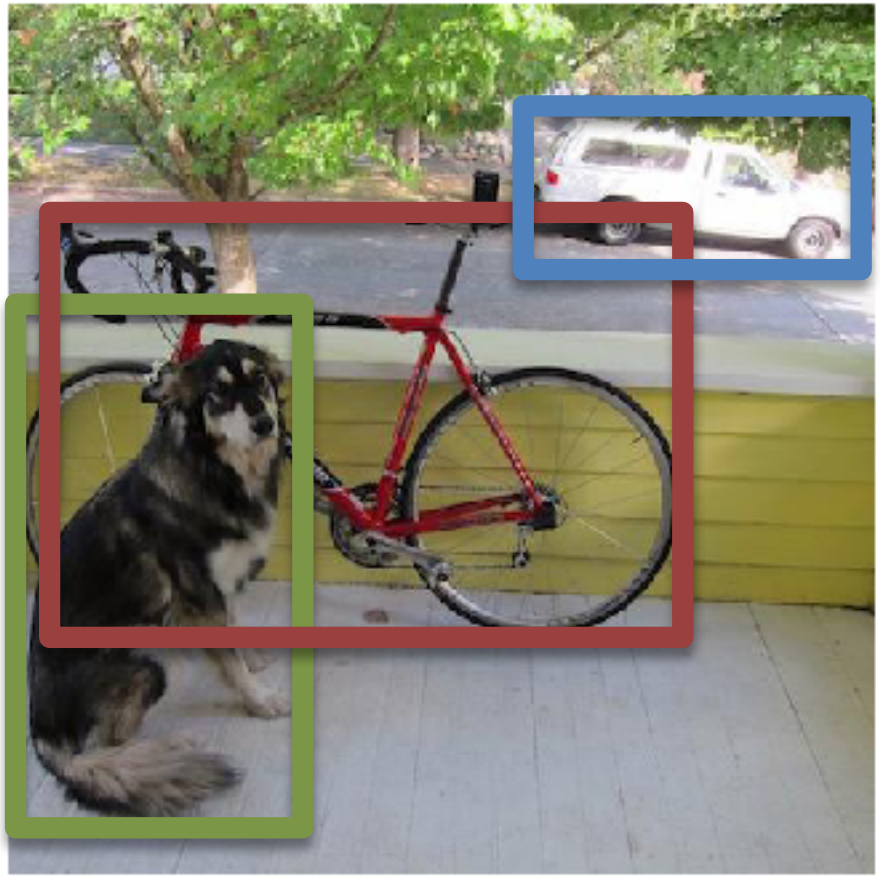
YOLO: You Only Look Once



Li Liu et al. Deep Learning for Generic Object Detection: A Survey 2018

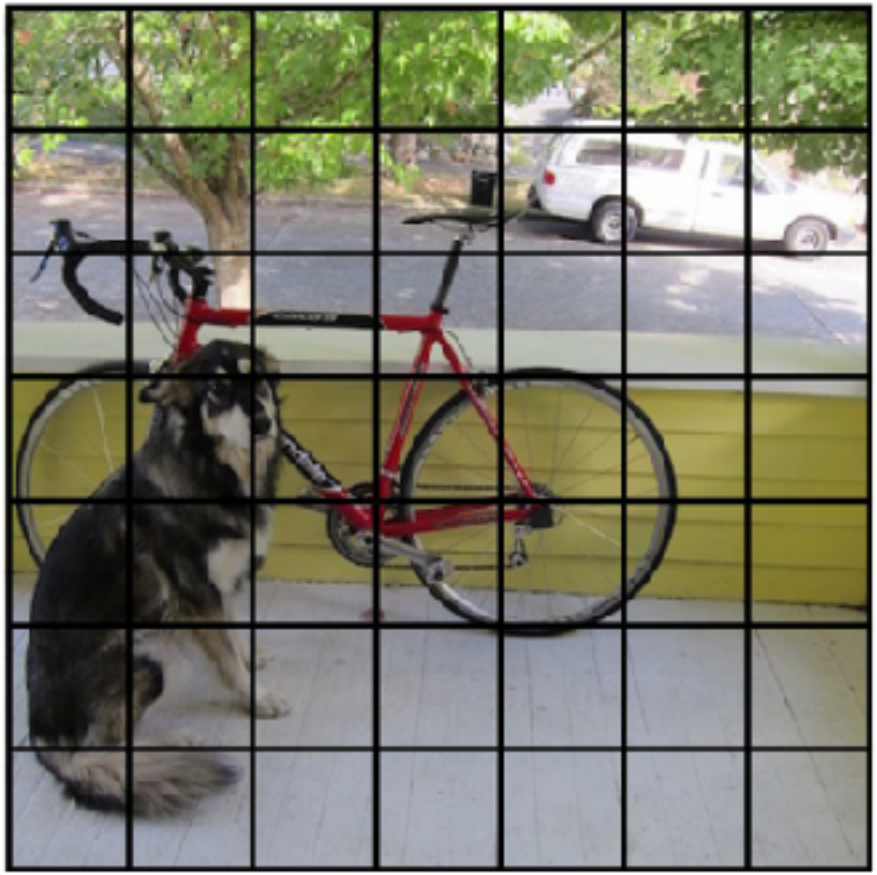
YOLO: You Only Look Once

- Detection Procedure



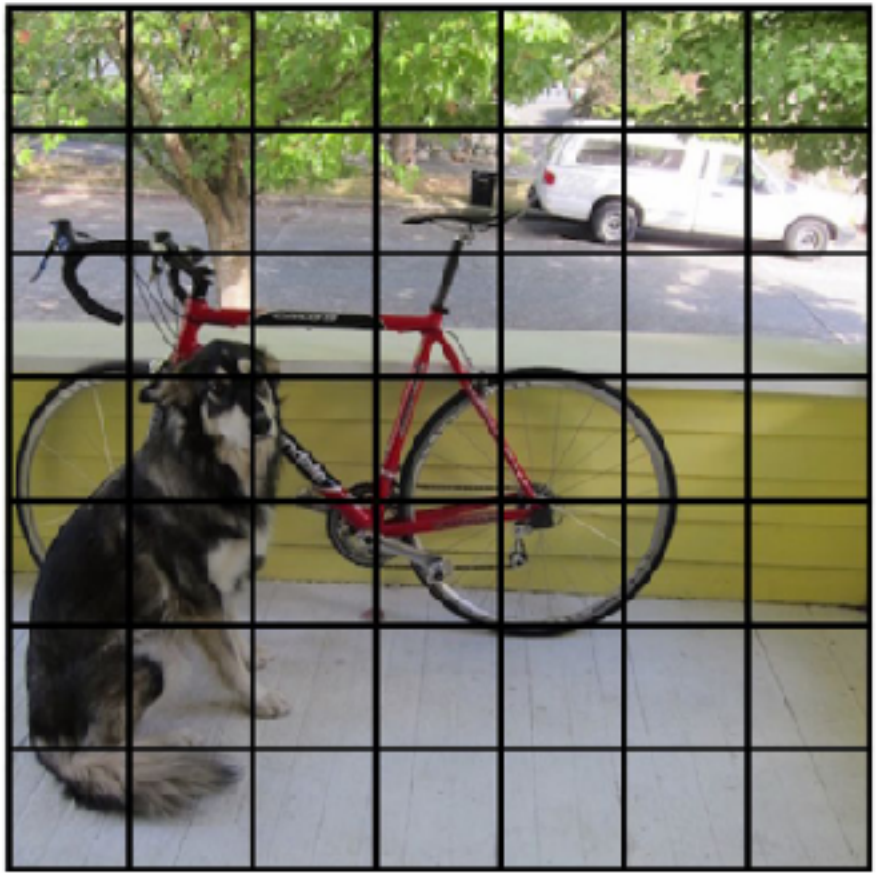
YOLO: You Only Look Once

We split the image into an S*S grid



YOLO: You Only Look Once

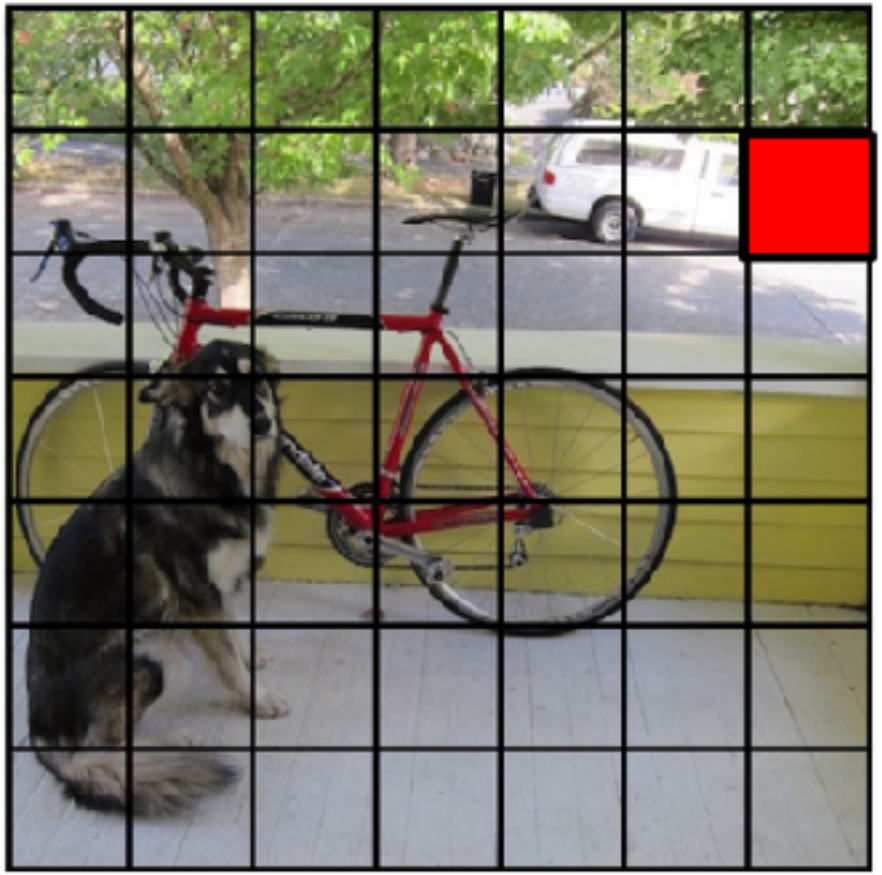
We split the image into an S*S grid



7*7 grid

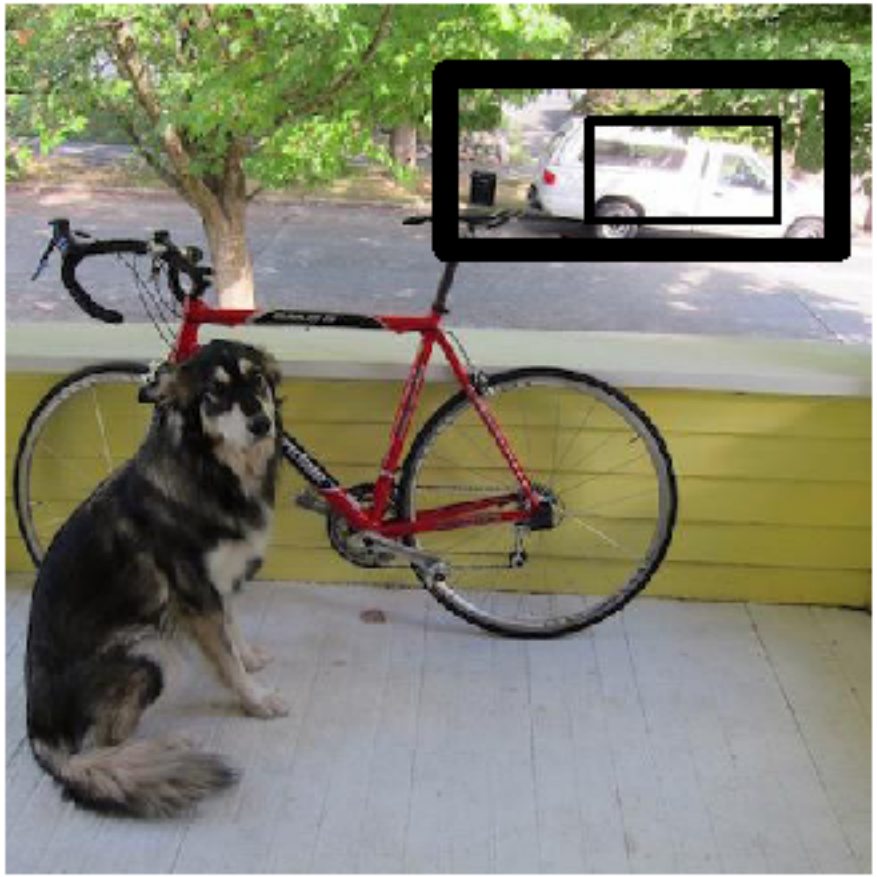
YOLO: You Only Look Once

Each cell predicts B boxes(x,y,w,h) and confidences of each box: P(Object)



YOLO: You Only Look Once

Each cell predicts B boxes(x,y,w,h) and confidences of each box: P(Object)

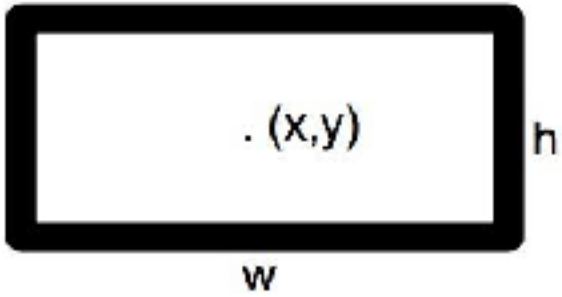


YOLO: You Only Look Once

Each cell predicts B boxes(x,y,w,h) and confidences of each box: P(Object)

B = 2

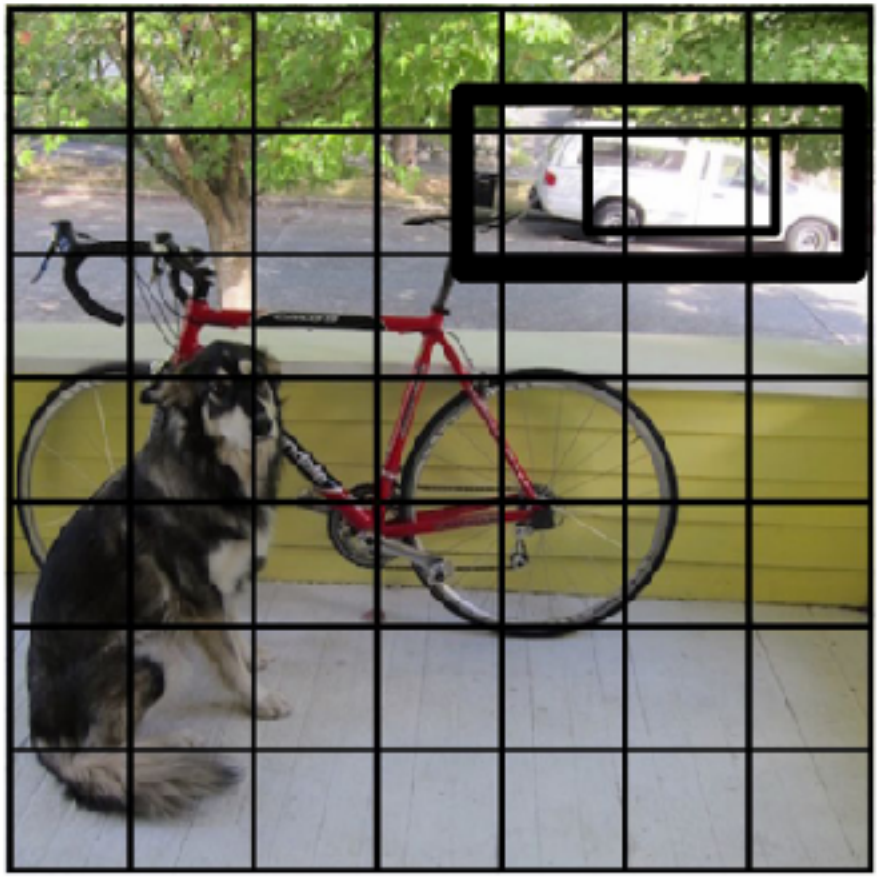
each box predict:



P(Object): probability that the box contains an object

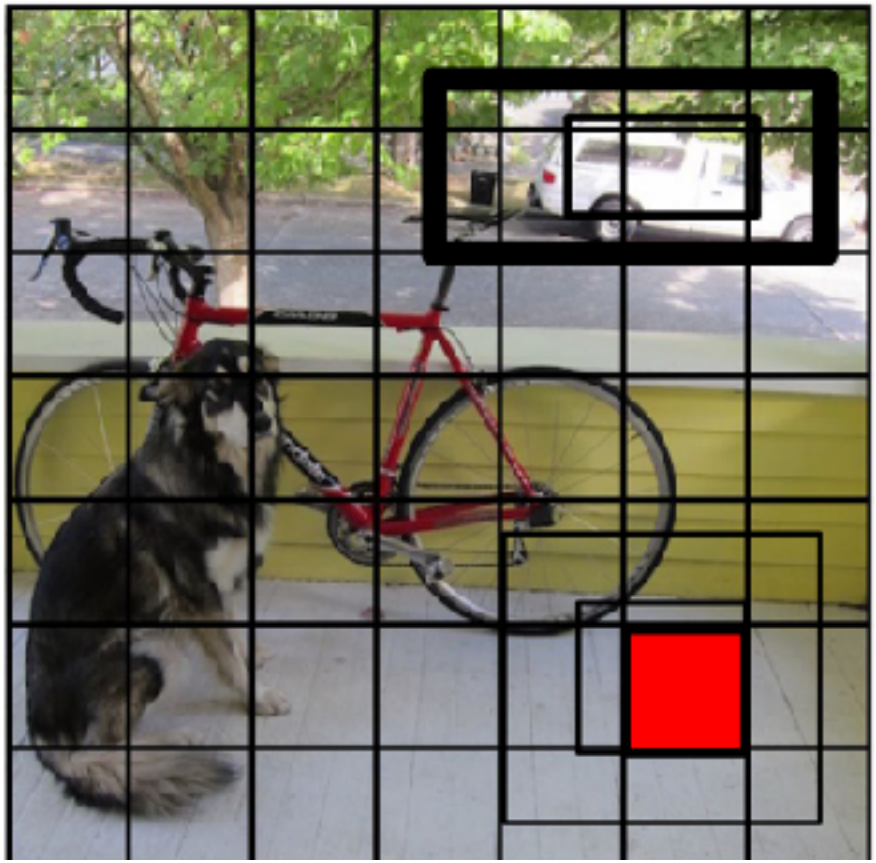
YOLO: You Only Look Once

Each cell predicts B boxes(x,y,w,h) and confidences of each box: P(Object)



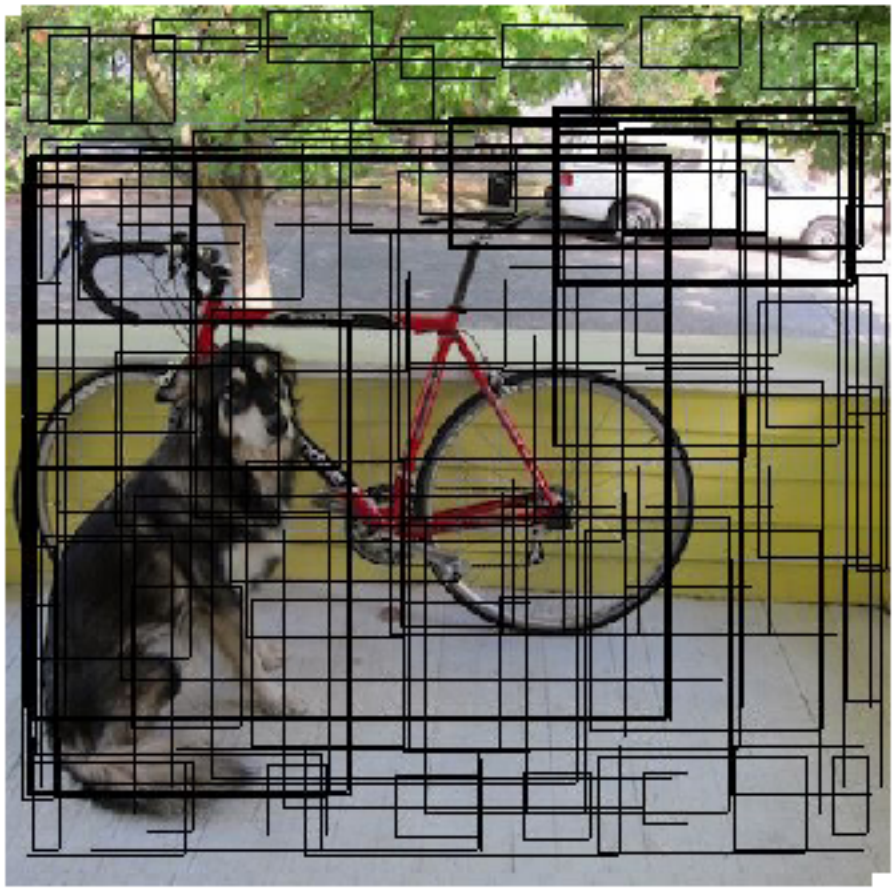
YOLO: You Only Look Once

Each cell predicts B boxes(x,y,w,h) and confidences of each box: P(Object)



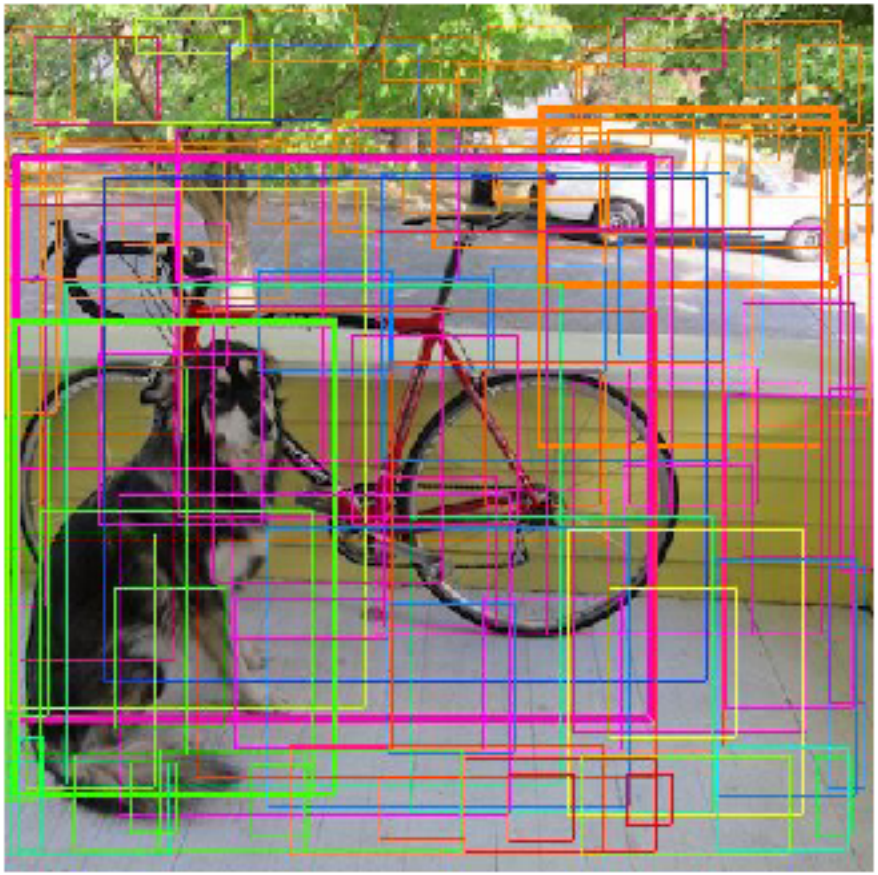
YOLO: You Only Look Once

Each cell predicts boxes and confidences: P(Object)



YOLO: You Only Look Once

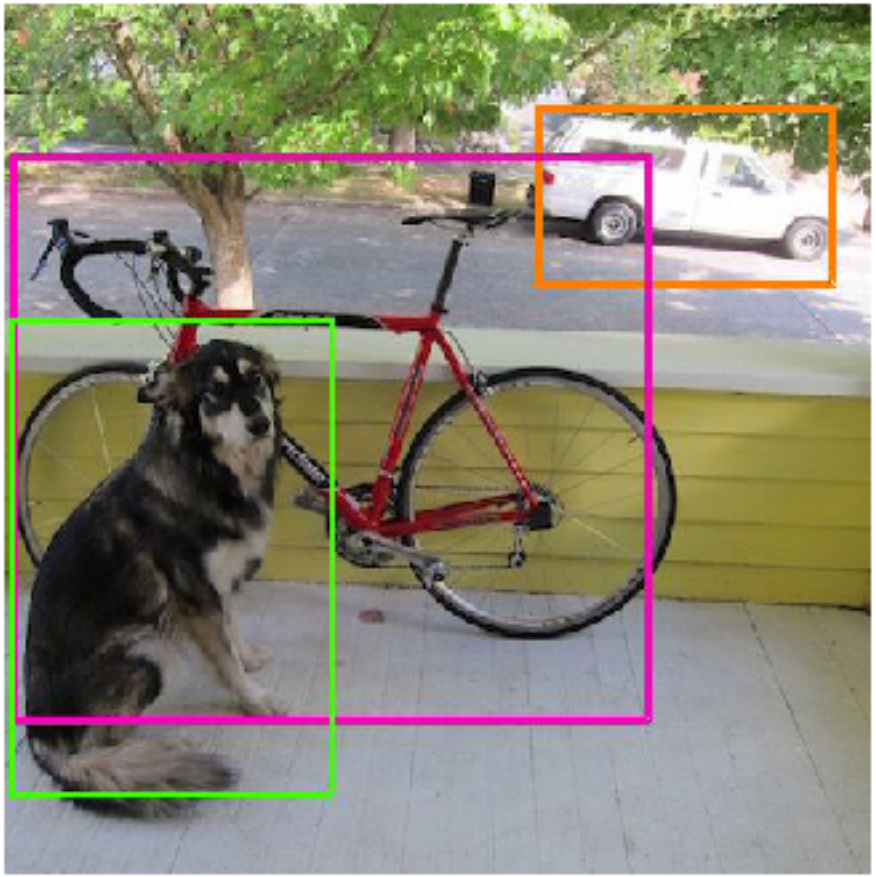
Then we combine the box and class predictions.



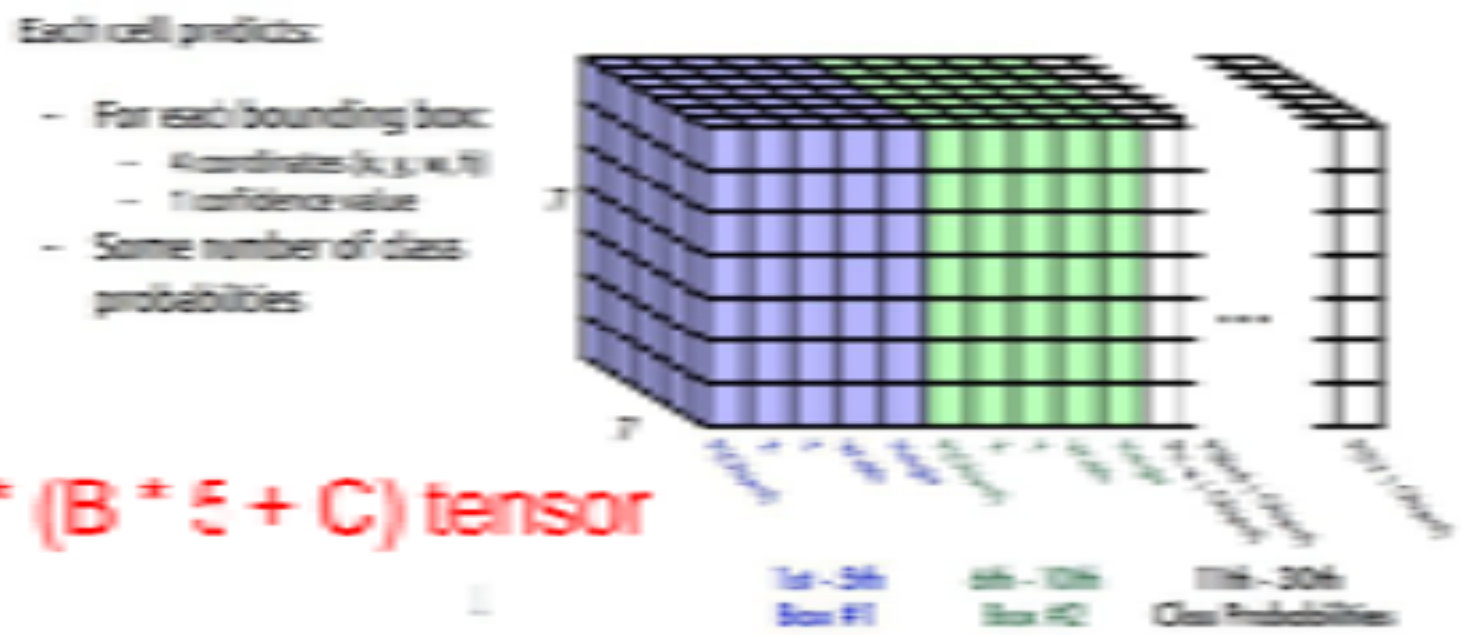
P(class|Object) * P(Object)
=P(class)

YOLO: You Only Look Once

Finally we do threshold detections and NMS



YOLO: You Only Look Once



$S * S * (B * 5 + C)$ tensor

Image Classification

Is this a dog or a person?

Neural Network Output

Dog = 1
Person = 0

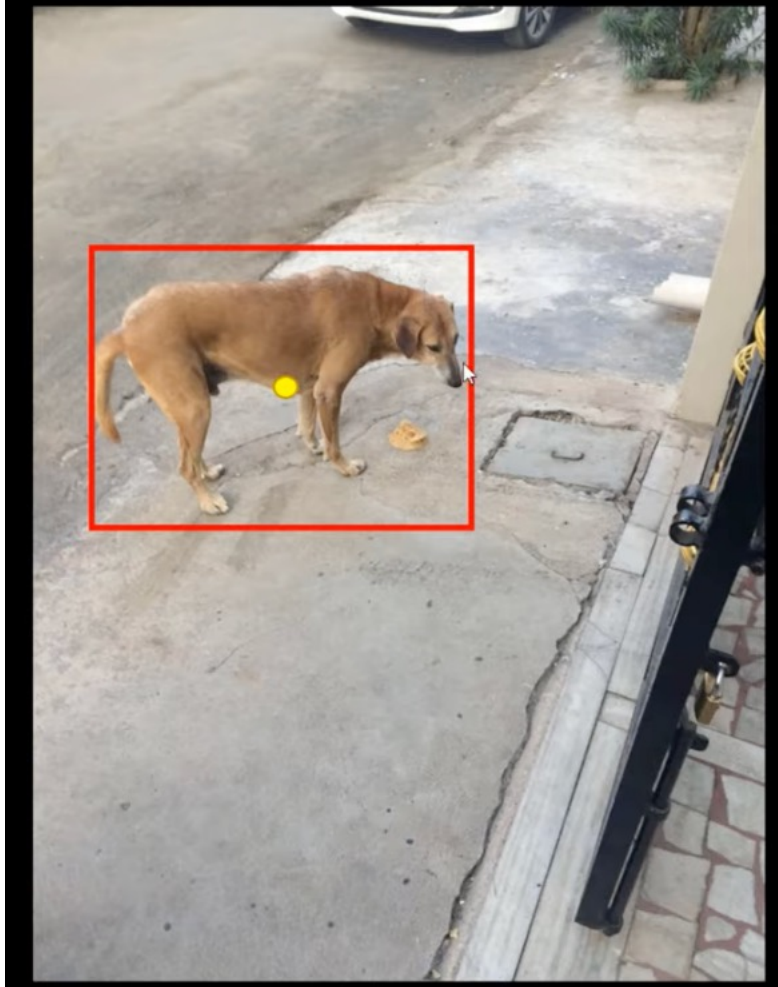
Object Localization

Where exactly is the dog in this image?

Neural Network Output

Dog = 1
Person = 0
+
Bounding Box

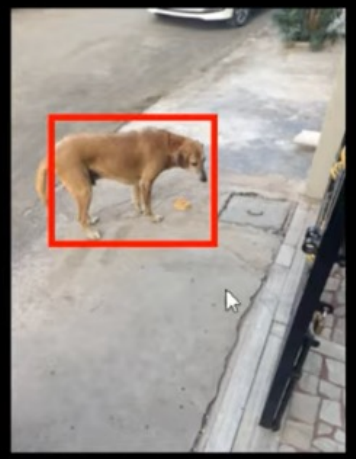
Object Localization



$$\begin{bmatrix} P_c \\ B_x \\ B_y \\ B_w \\ B_h \\ C_1 \\ C_2 \end{bmatrix} \quad \begin{bmatrix} 1 \\ 50 \\ 70 \\ 60 \\ 70 \\ 1 \\ 0 \end{bmatrix}$$

C_1 = *Dog class*
C2 = Person Class

X_train

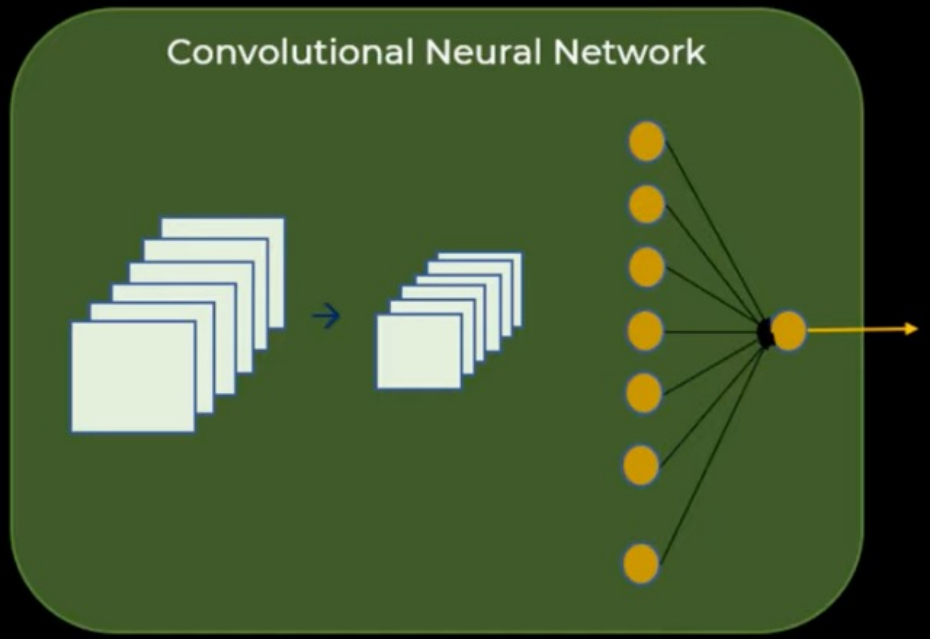


y_train

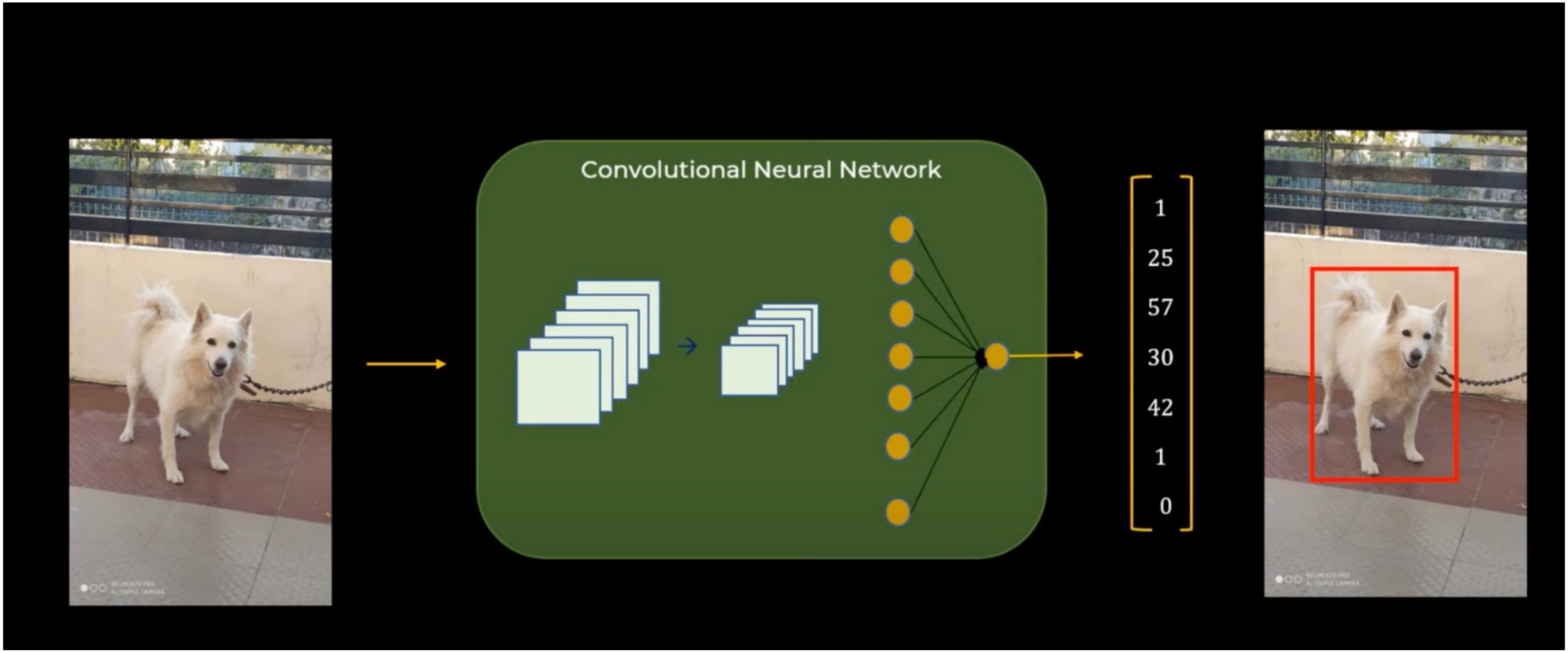
$$\begin{bmatrix} P_c \\ B_x \\ B_y \\ B_w \\ B_h \\ C_1 \\ C_2 \end{bmatrix} \quad \begin{bmatrix} 1 \\ 50 \\ 70 \\ 60 \\ 70 \\ 1 \\ 0 \end{bmatrix}$$

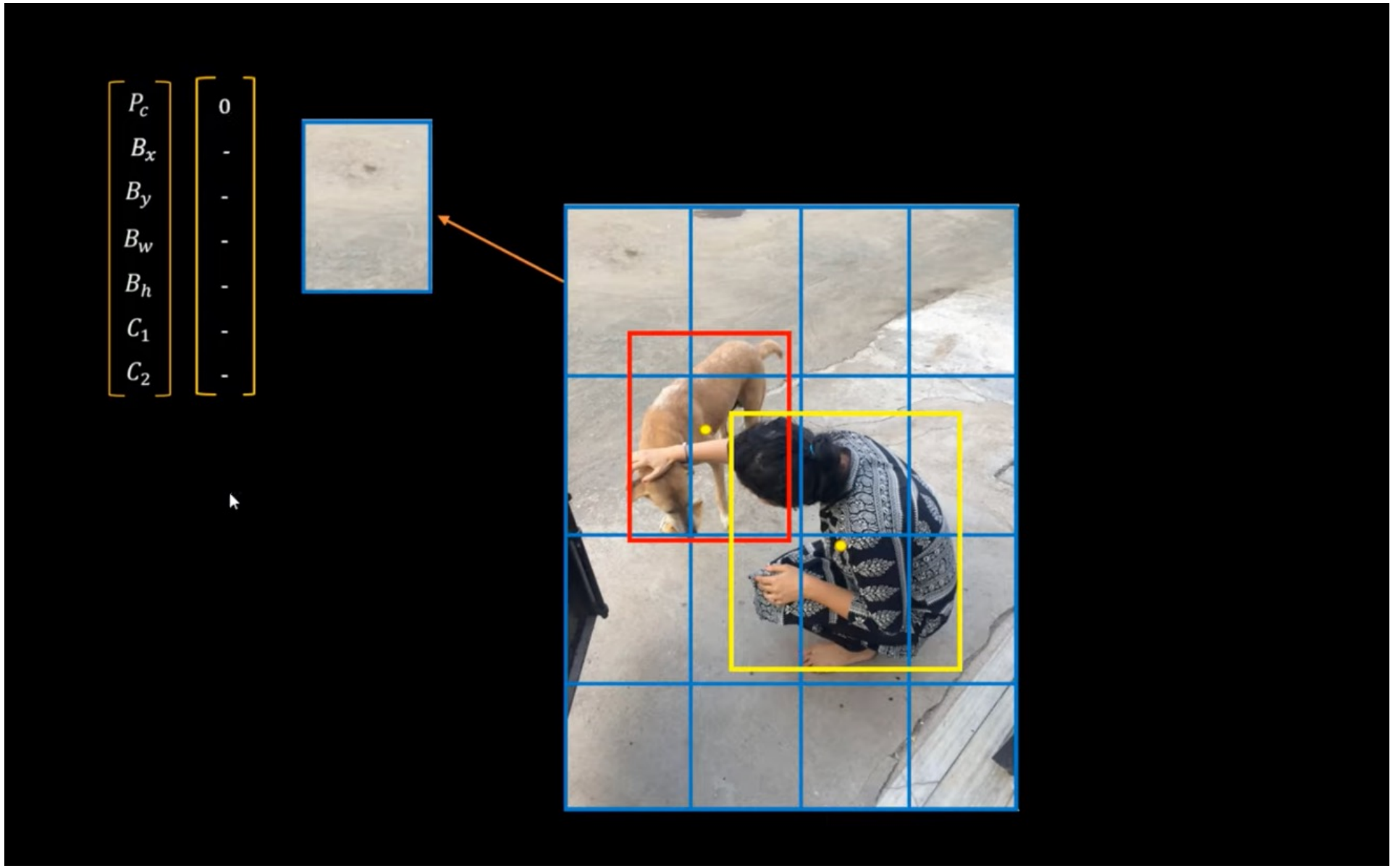
$$\begin{bmatrix} 1 \\ 30 \\ 55 \\ 28 \\ 82 \\ 0 \\ 1 \end{bmatrix}$$

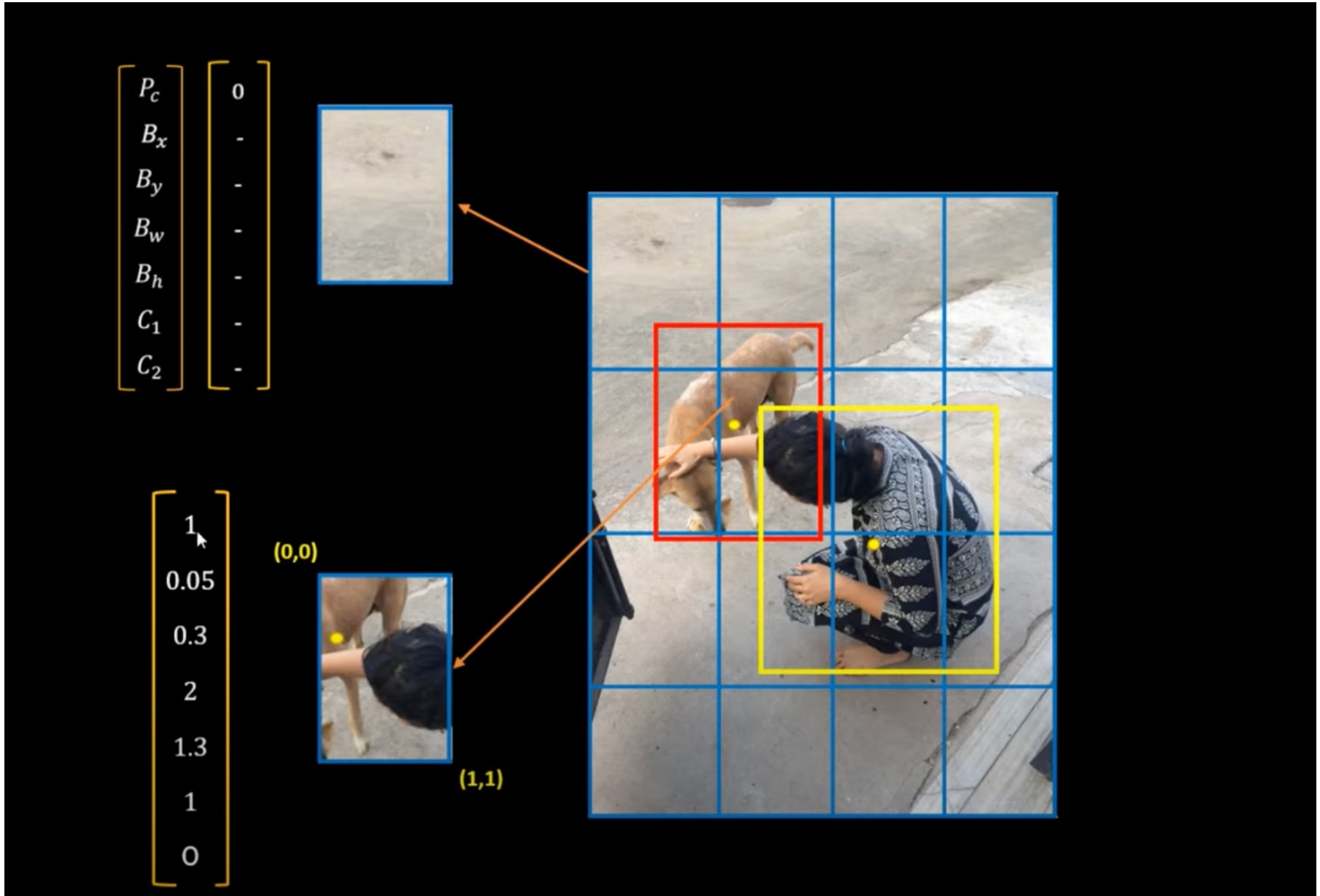
$$\begin{bmatrix} 0 \\ - \\ - \\ - \\ - \\ - \\ - \end{bmatrix}$$

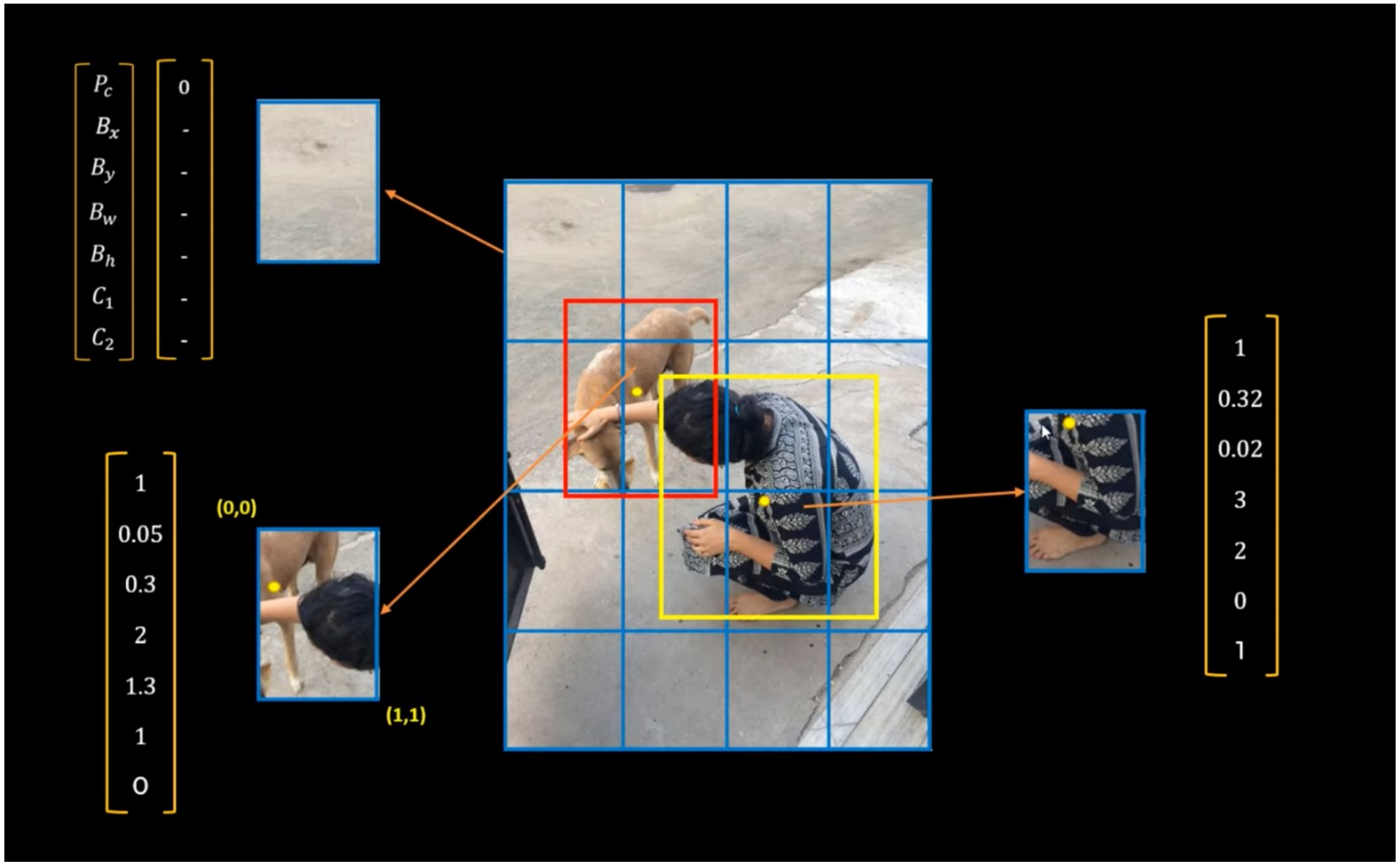


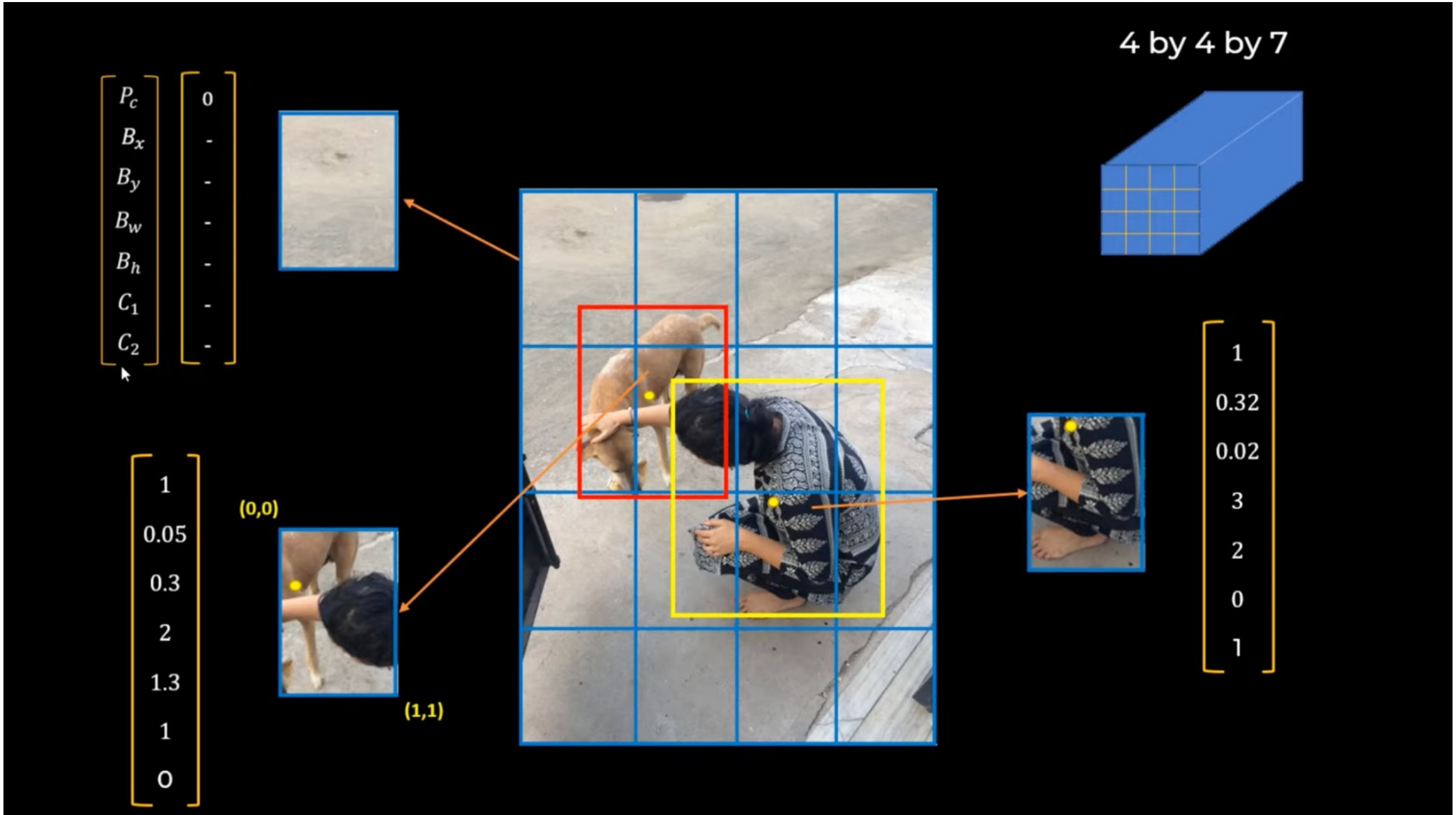
$$\begin{bmatrix} P_c \\ B_x \\ B_y \\ B_w \\ B_h \\ C_1 \\ C_2 \end{bmatrix}$$

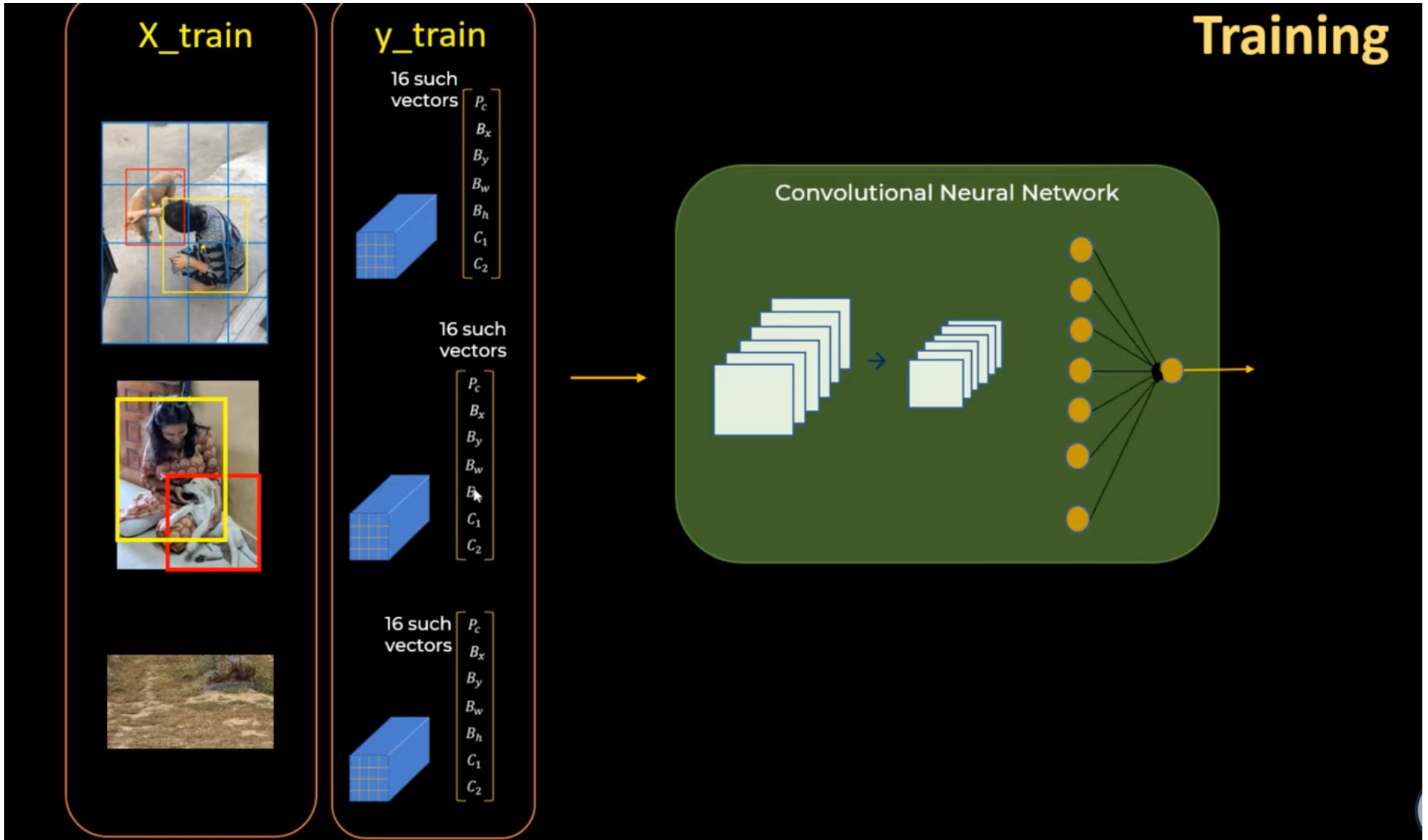

Convolutional Neural Network
1
25
57
30
42
1
0


P_c
B_x
B_y
B_w
B_h
C_1
C_2
0
-
-
-
-
-
-

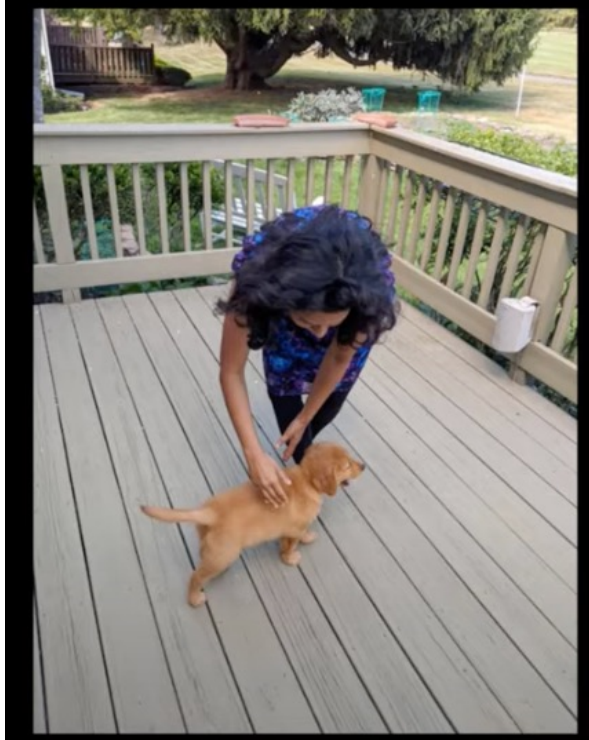

P_c
B_x
B_y
B_w
B_h
C_1
C_2
0
-
-
-
-
-
-
1
0.05
0.3
2
1.3
1
0
(0,0)
(1,1)


P_c
B_x
B_y
B_w
B_h
C_1
C_2
0
-
-
-
-
-
-
1
0.05
0.3
2
1.3
1
0
(0,0)
(1,1)
1
0.32
0.02
3
2
0
1


4 by 4 by 7
P_c
B_x
B_y
B_w
B_h
C_1
C_2
0
-
-
-
-
-
-
1
0.05
0.3
2
1.3
1
0
(0,0)
(1,1)
1
0.32
0.02
3
2
0
1


X_train
y_train
Training
16 such vectors
P_c
B_x
B_y
B_w
B_h
C_1
C_2
16 such vectors
16 such vectors
Convolutional Neural Network

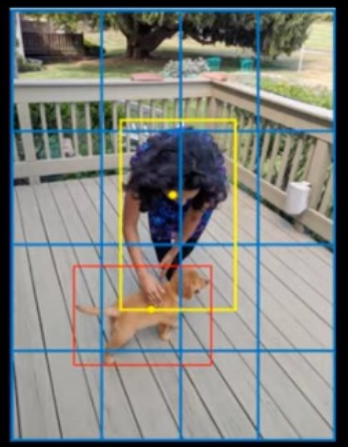
Prediction



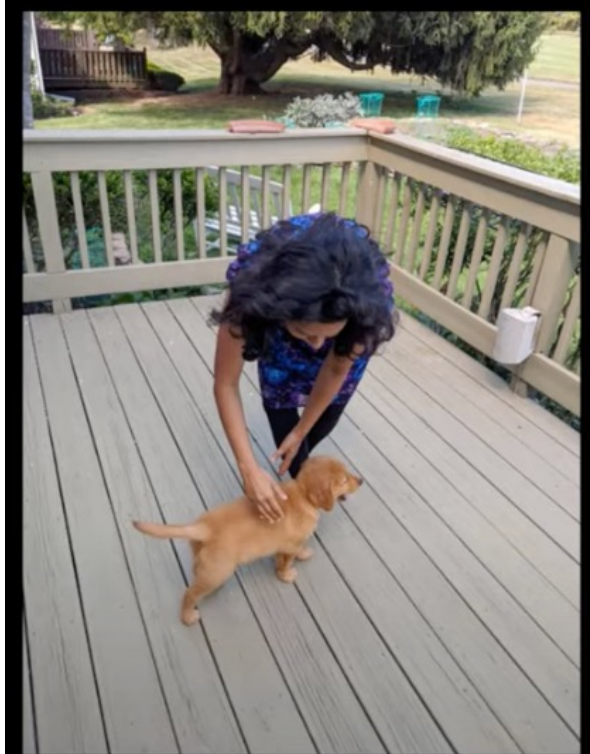
Convolutional Neural Network

16 such vectors

$$\begin{bmatrix} P_c \\ B_x \\ B_y \\ B_w \\ B_h \\ C_1 \\ C_2 \end{bmatrix}$$



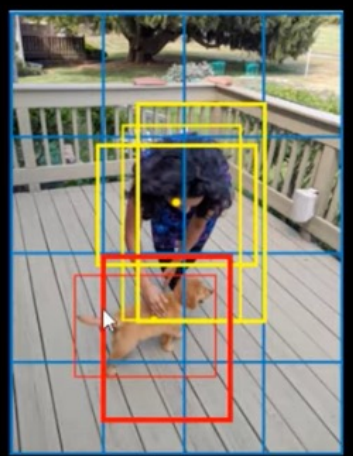
Prediction



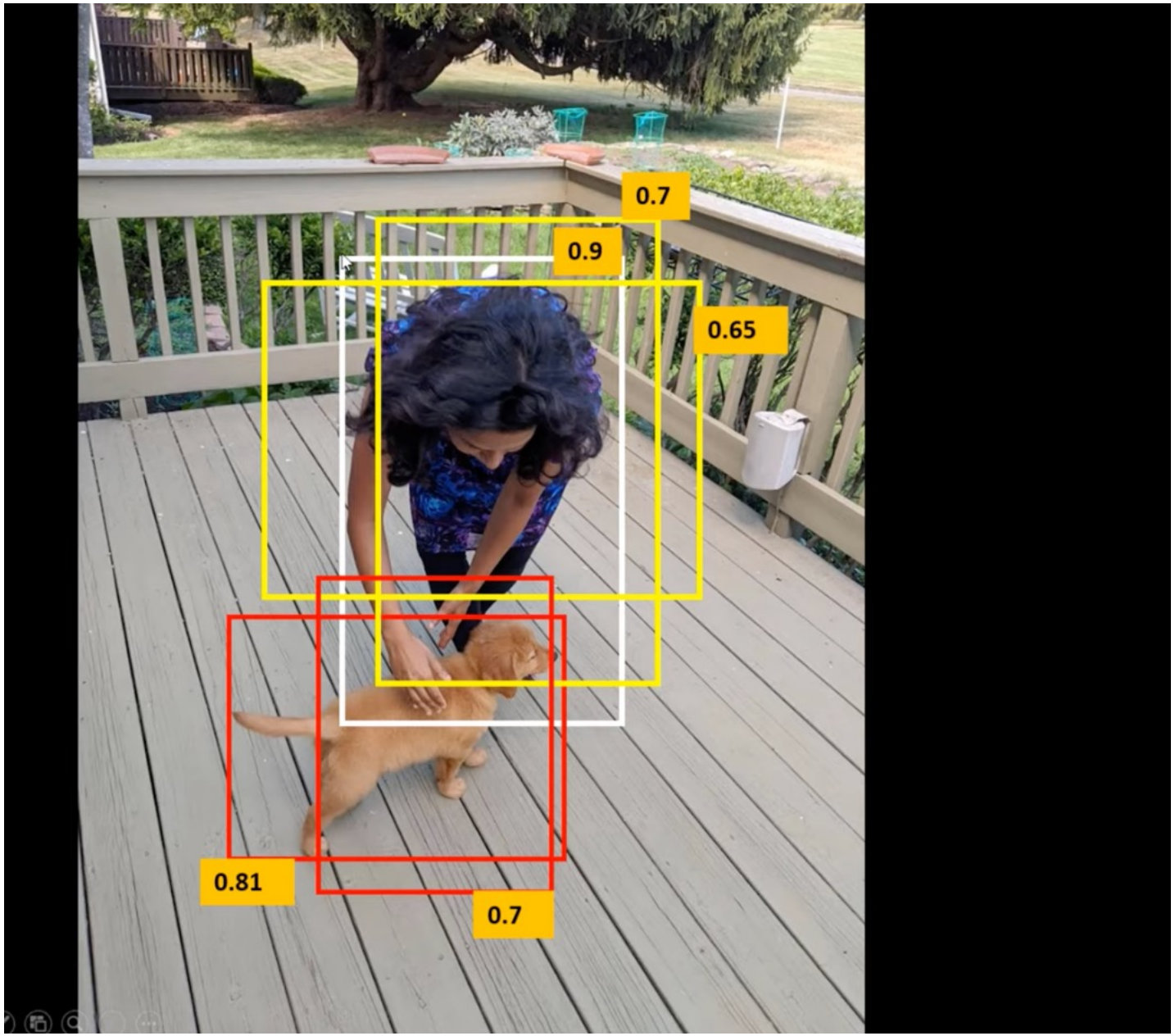
Convolutional Neural Network

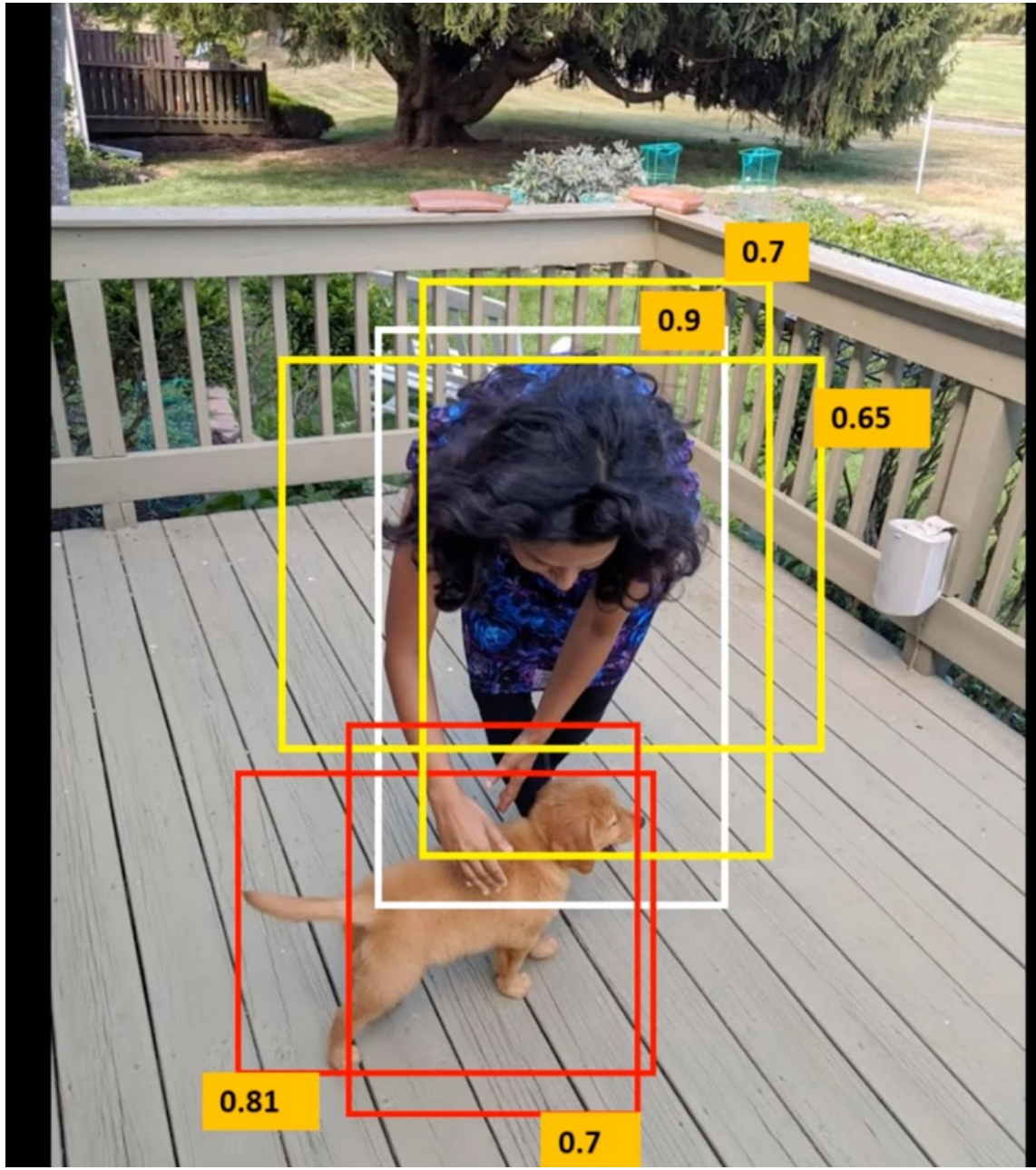
16 such vectors

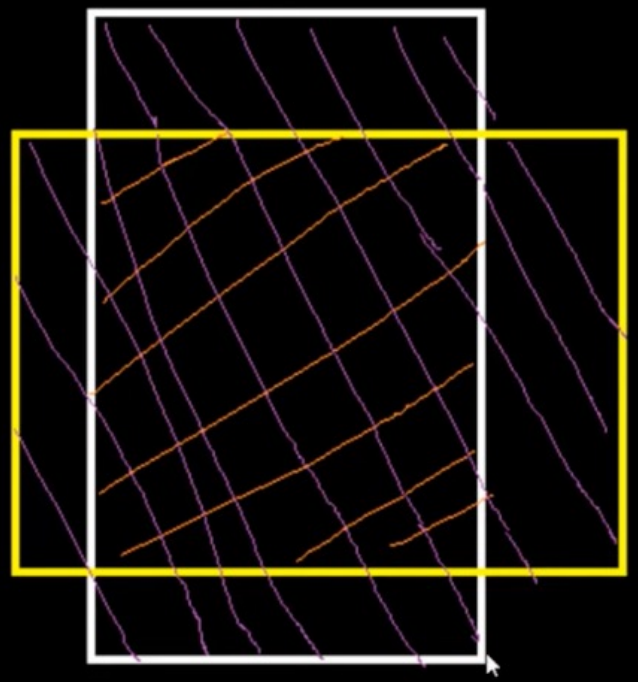
$$\begin{bmatrix} P_c \\ B_x \\ B_y \\ B_w \\ B_h \\ C_1 \\ C_2 \end{bmatrix}$$



Multiple bounding boxes

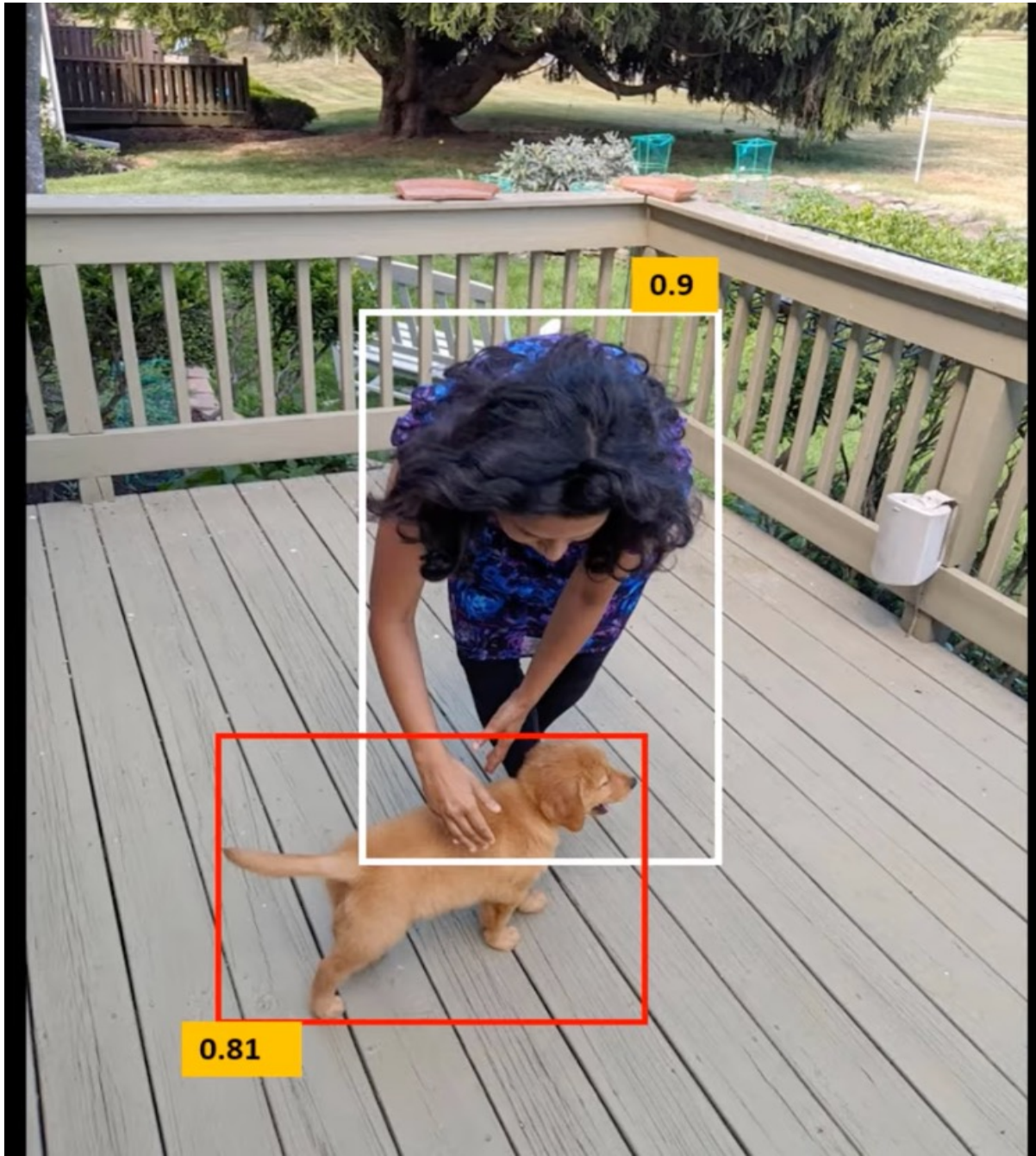

0.7
0.9
0.65
0.81
0.7





Intersection over union = intersect area / union area

Intersection over union : IOU



Non max suppression

YOLO: You Only Look Once

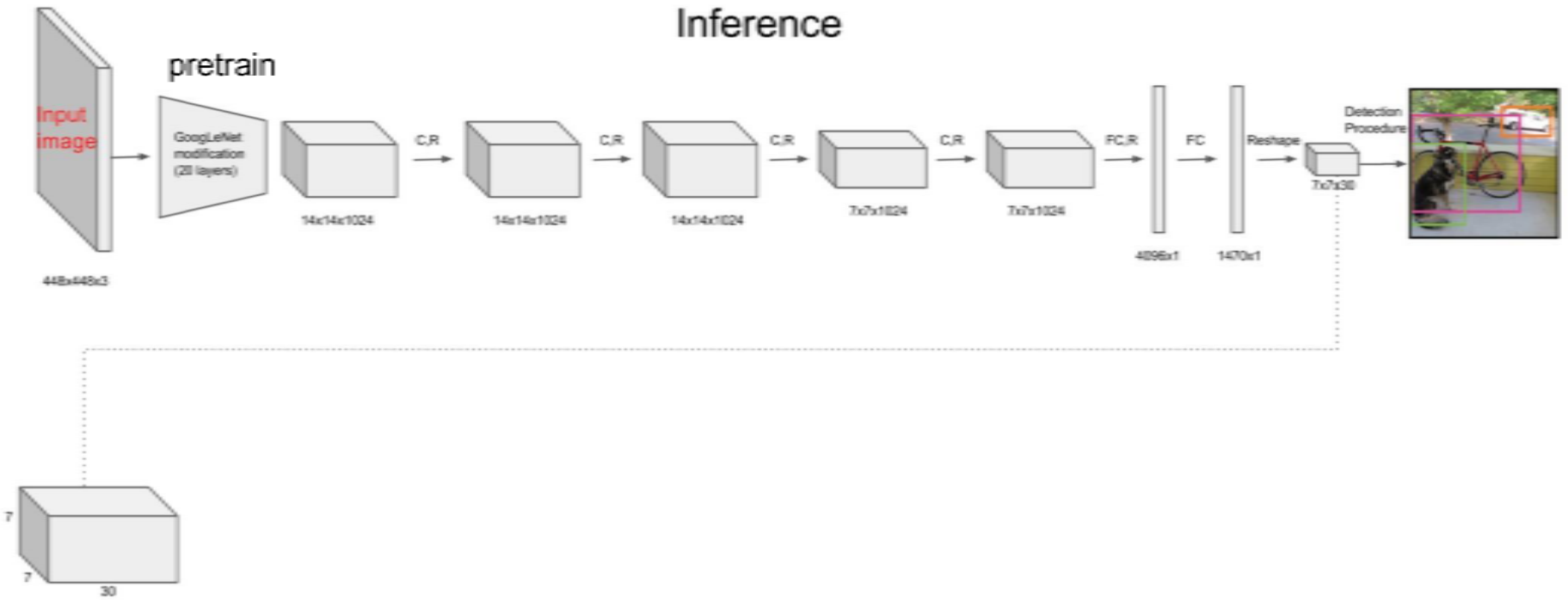
Network

YOLO: You Only Look Once

https://zhuanlan.zhihu.com/p/24916786?refer=xiaoleimlnote

YOLO: You Only Look Once

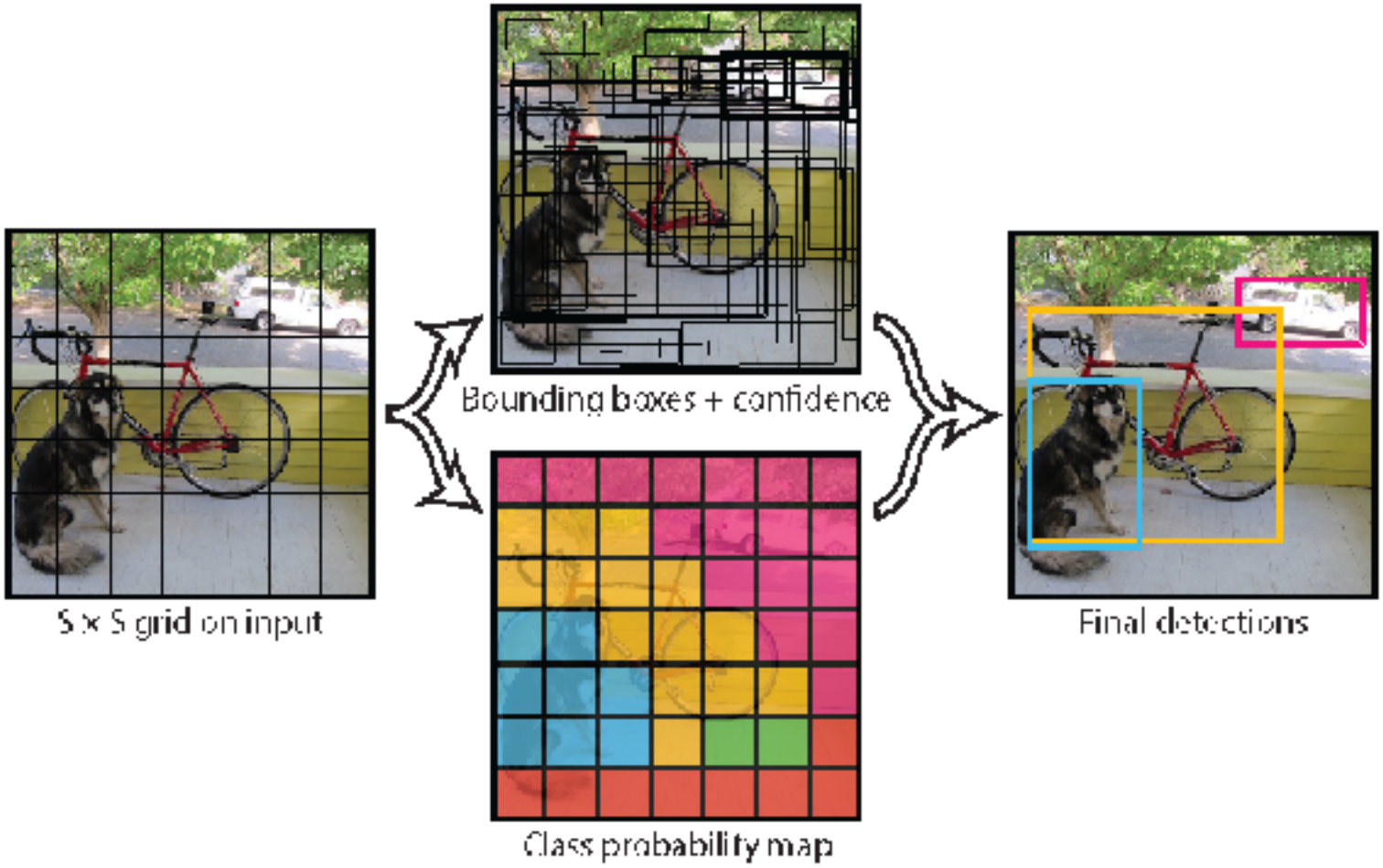
https://zhuanlan.zhihu.com/p/24916786?refer=xiaoleimlnote



YOLO: Loss Function

- $\mathbb{1}_i^{obj}$ is 1 if there is an object in cell i and 0 otherwise,
- $\mathbb{1}_{i,j}^{obj}$ is 1 if there is an object in cell i and predicted box j is the most fitting one, 0 otherwise
- $p_{i,c}$ is 1 if there is an object of class c in cell i, and 0 otherwise,
- x_i, y_i, w_i, h_i the annotated object bounding box (defined only if $\mathbb{1}_i^{obj} = 1$, and relative in location and scale to the cell),
- $c_{i,j}$ IOU between the predicted box and the ground truth target.

YOLO: You Only Look Once



(Redmon et al., 2015)