April 7, 2023

DSML: Computer Vision.

Object Localization and Detection - 2

Class starts @ 9:05 pm.

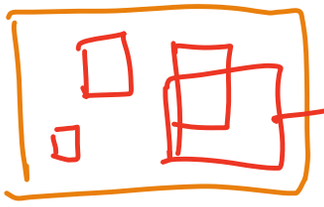
Recap:

* Bounding boxes

1] $(x_s, y_s), (x_e, y_e)$ 2] (x_c, y_c, w, h).

* Output of an object detection network.
  - ① C - class of the object detected, → classification ✓
  - (2-5) B. Box co-ordinates. → Regression. ✓

* How to deal with multiple bounding box proposals?



→ 20,000 guesses.

* 1 network to evaluate them all!!

20

* Non-max suppression (NMS).

I.o.U - Intersection over Unions to check if boxes are representing the same object or not.

Agenda:

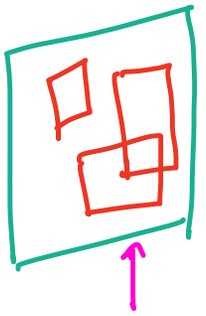
* Two stage methods - R-CNN family.
  → Slow, but high accuracy.

* Single stage methods - YOLO family.
  → Faster, slightly lower accuracy.

* Real time application - Self driving cars.

R-CNN family :

* Naive :

① 

② Sliding window to get region proposals.

Maybe no CNN here.

③ Apply Image classification to all region proposals.

CNN.

↓ o/p

④ Confidence for each proposal.

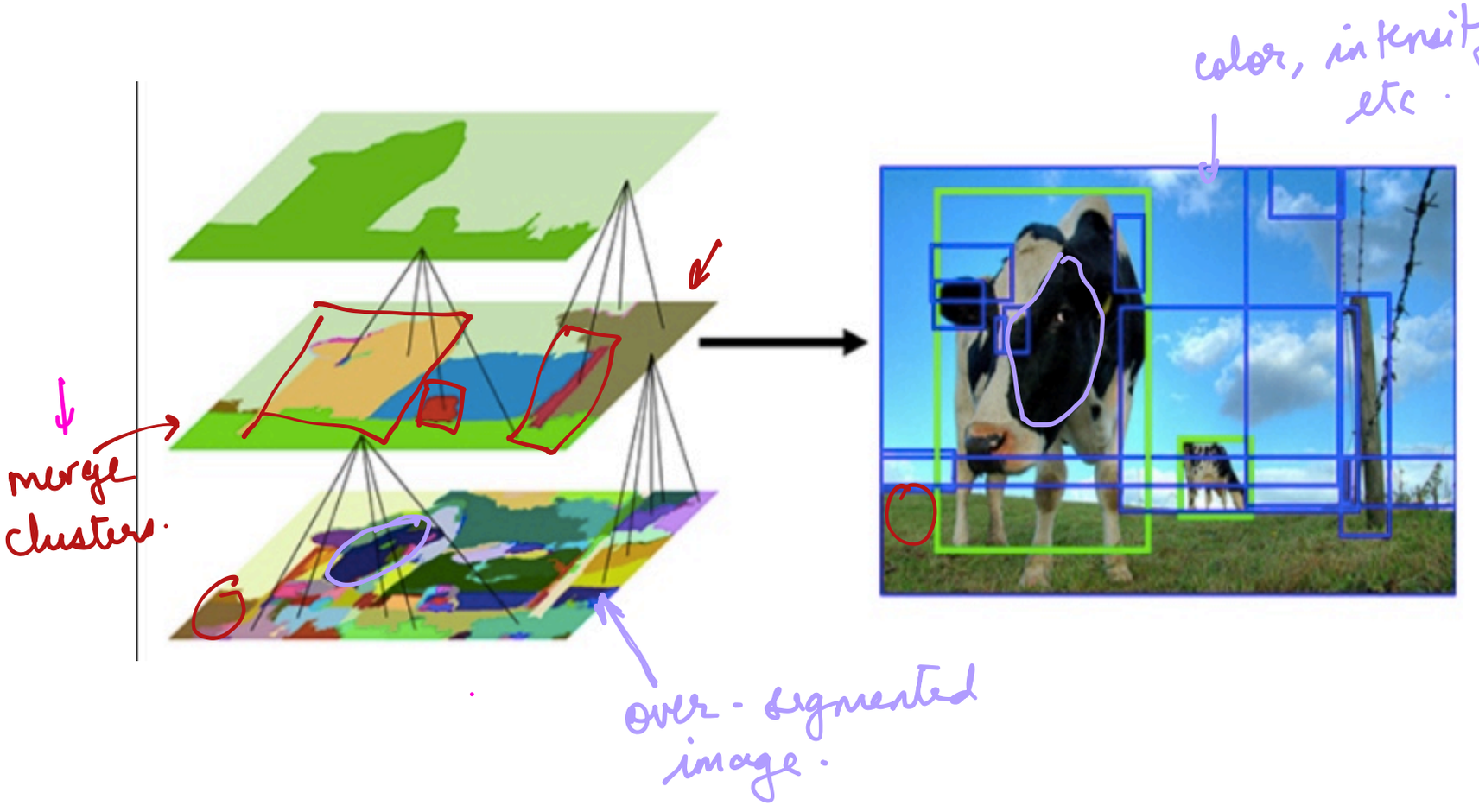
CNN.

⑤ Only keep high confidence bounding boxes.

R CNN - 2014 → ② and instead of using all possible shifts & scales of window, use selective search to extract proposals

(Imagenet 2013 detection dataset).

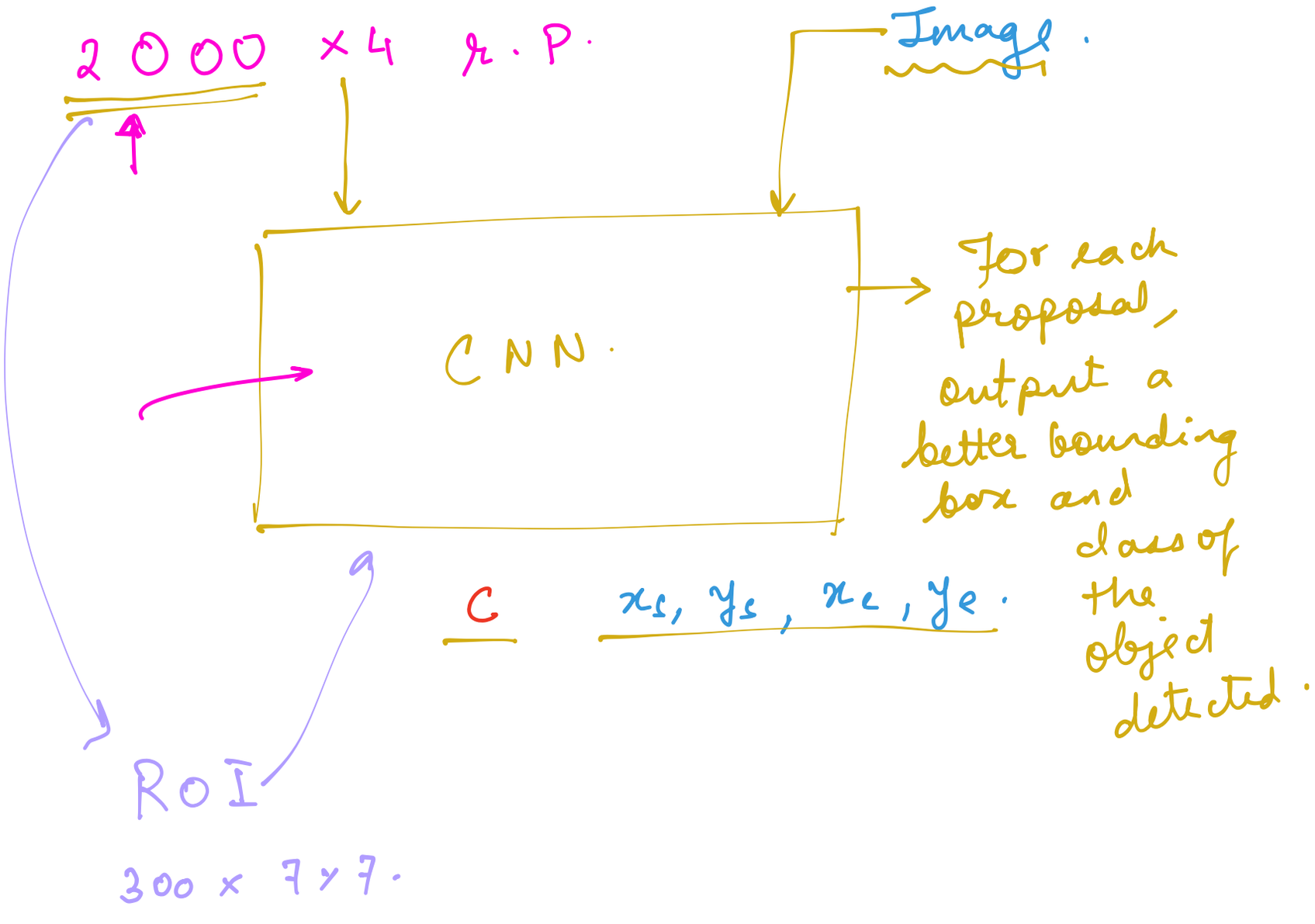
Selective Search instead of all possibilities:



"Region proposal algorithm" ≈ 2000 BB.

(2000 × 4.)

Important because it is much much faster than step 2 in the naive algorithm.


2000 x 4 r.p.
Image.
CNN.
For each proposal, output a better bounding box and class of the object detected.
C
x_s, y_s, x_e, y_e.
RoI
300 x 7 x 7.

External region proposals method (Selective Search Algorithm)	✓
	R-CNN
Test time per image	50 seconds
~~Speed-up~~	~~1x~~
mAP (VOC 2007)	66.0%

→ Not practical at all!!

(a) Clustering/S.S. takes time.

(b) We have to run a CNN 2000 times per image!!

major bottleneck.

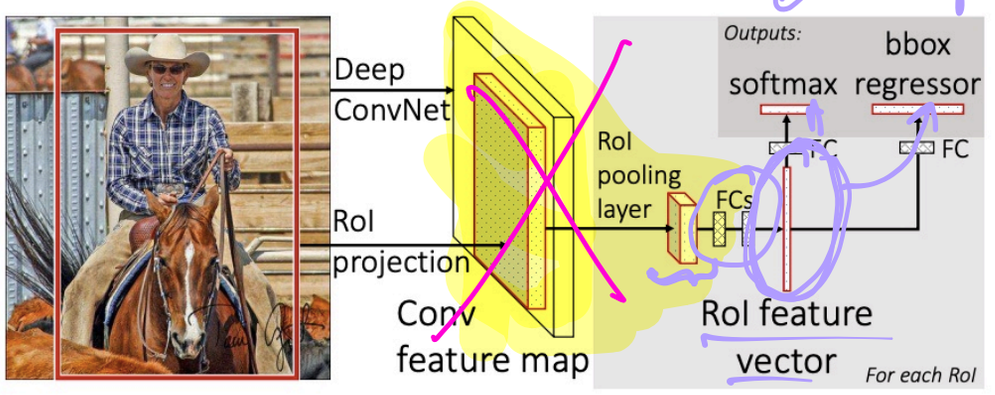
mAP → Mean Average Precision.

Fast R-CNN.

* The bottleneck seems to be 1000 proposals per image.

* Solution: Add a ROI pooling algorithm.

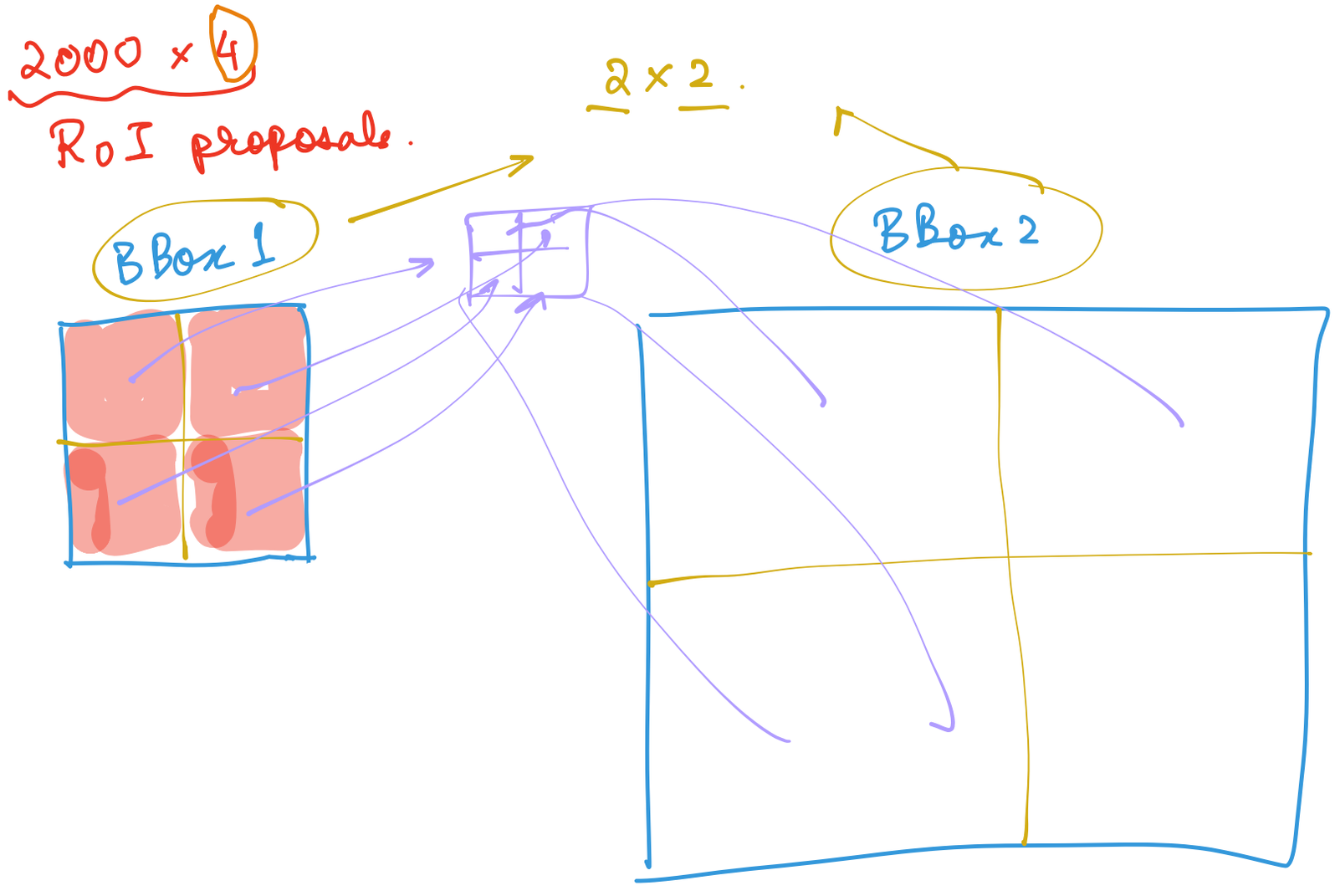
CNN architecture.



refined B.B.

2000 ↓ ≈ 300

Figure 1. Fast R-CNN architecture. An input image and multiple regions of interest (RoIs) are input into a fully convolutional network. Each RoI is pooled into a fixed-size feature map and then mapped to a feature vector by fully connected layers (FCs). The network has two output vectors per RoI: softmax probabilities and per-class bounding-box regression offsets. The architecture is trained end-to-end with a multi-task loss.

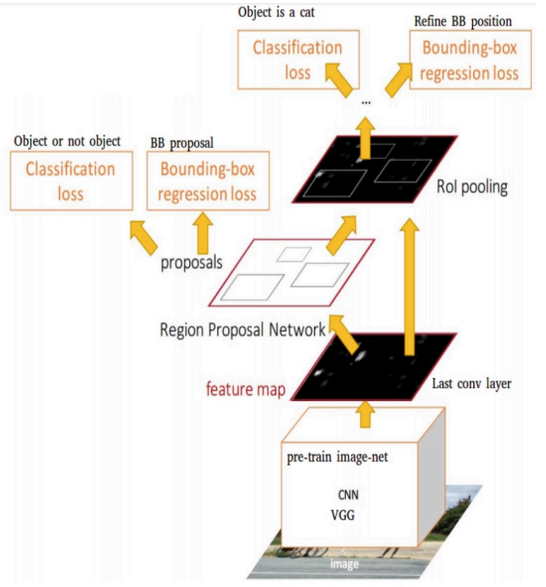

2000 × 4
RoI proposals.
2 × 2.
BBox 1
BBox 2

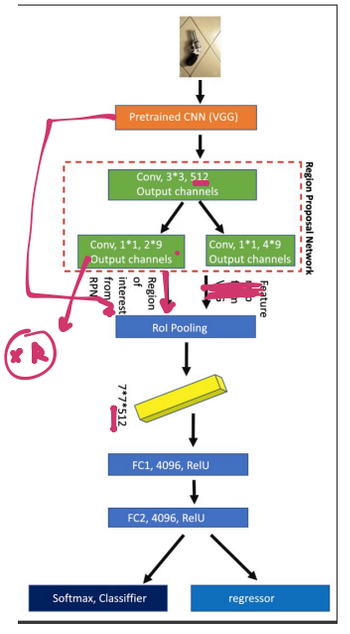
External region proposals method (Selective Search Algorithm)	✓	✓
	R-CNN	**Fast R-CNN**
Test time per image	50 seconds	2 seconds
Speed-up	1x	25x
mAP (VOC 2007)	66.0%	66.9%

Faster R-CNN paper:

→ These guys introduced a CNN which does the job of the S.S./Clustering algorithm.

→ RPN - region proposal network.





External region proposals method (Selective Search Algorithm)	✓	✓	✗
	R-CNN	**Fast R-CNN**	**Faster R-CNN**
Test time per image	50 seconds	2 seconds	0.2 seconds
Speed-up	1x	25x	250x
mAP (VOC 2007)	66.0%	66.9%	66.9%

1 — 0.2 seconds.

? — 1 second?

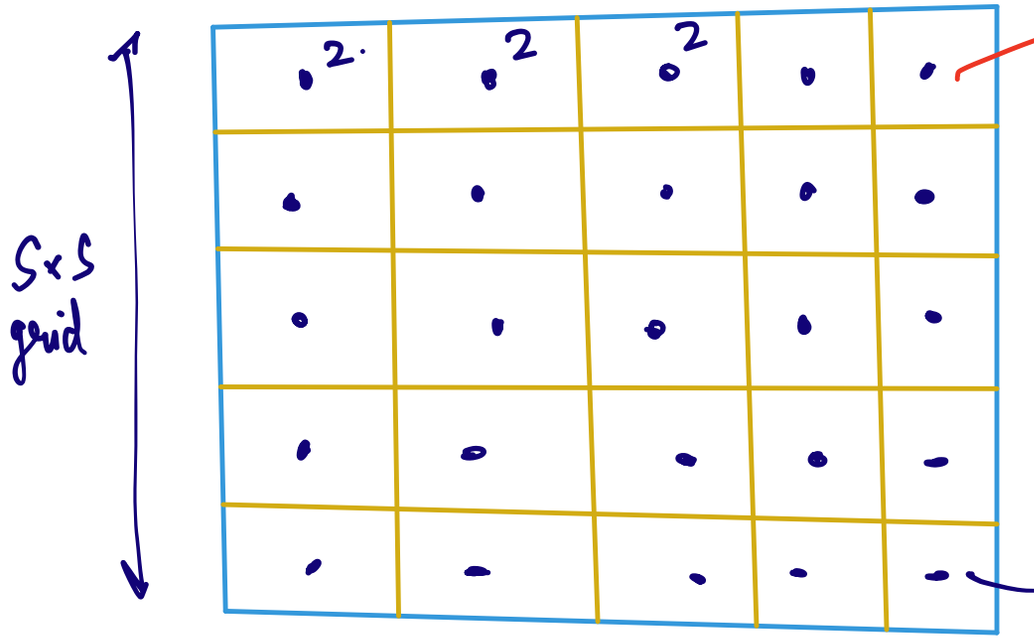
$\frac{1}{0.2} = 5$

Human visual system

↓

30 – 60 fps.

YOLO family → You only look once!



RCNN - look for proposals which can be anywhere in this image.

YOLO family: 2 proposals per cell.

YOLO v3 in easy steps :

YOLO in easy steps:

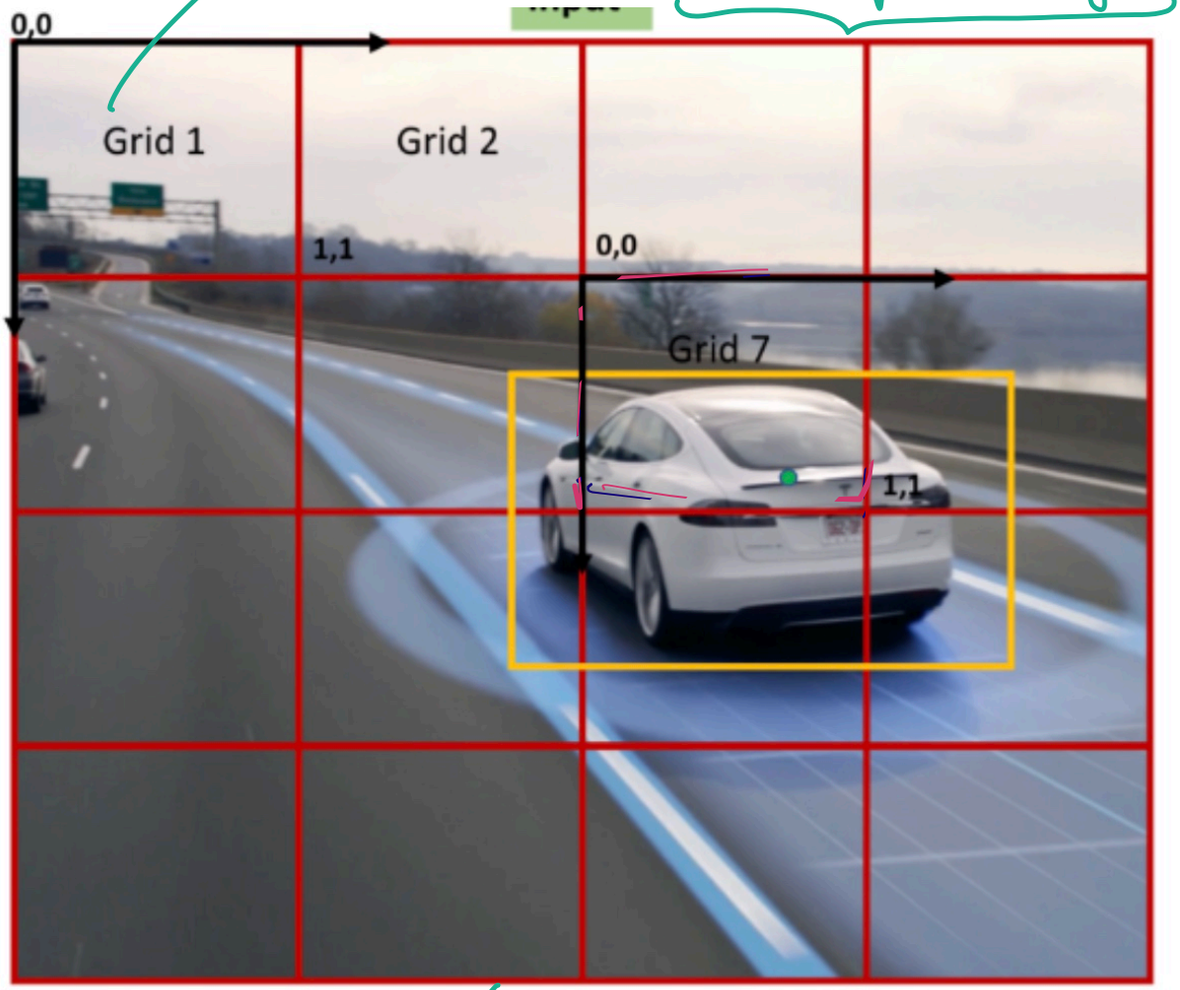
1. Divide the image into multiple grids. For illustration, I have drawn 4x4 grids in above figure.
2. Label the training data as shown in the above figure.
   - If **C is number of unique objects** in our data, `S*S is number of grids into which we split our image`, then our output vector will be of length `S*S*(C+5)`.
   - **For e.g.** in above case, our target vector is 4x4x(3+5) as we divided our images into 4*4 grids and are training for `3 unique objects: Car, Light and Pedestrian`.
3. Make one deep convolutional neural net with **loss function as error between output activations and label vector**.
   - Basically, `the model predicts the output of all the grids in just one forward pass of input image through ConvNet`.
4. **Keep in mind that the label for object being present in a grid cell (P.Object) is determined by the presence of object's centroid in that grid.**
   - This is important to not allow one object to be counted multiple times in different grids.

B → No. of region proposals per grid cell

CNN →

1. If object is there.
2. 4 co-ordinates.
3. Class of the object.

} Each proposal for that grid.



$1 \rightarrow B \times (5 + C) \rightarrow$ no. of classes present.

B → No. of object.

5 → BBox cordinate.

Input dataset

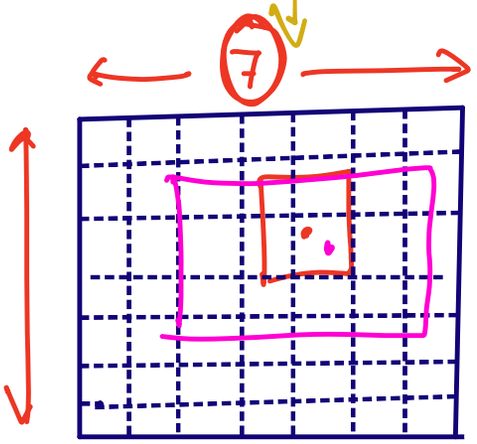
→ Images

→ Annotation.

} → Enough for RCNN.

But, it is not enough for YOLO.

Preprocessing.



1 class, 4 bb.

(5,)

1

0

B

(5)

B-hot encoded.

20

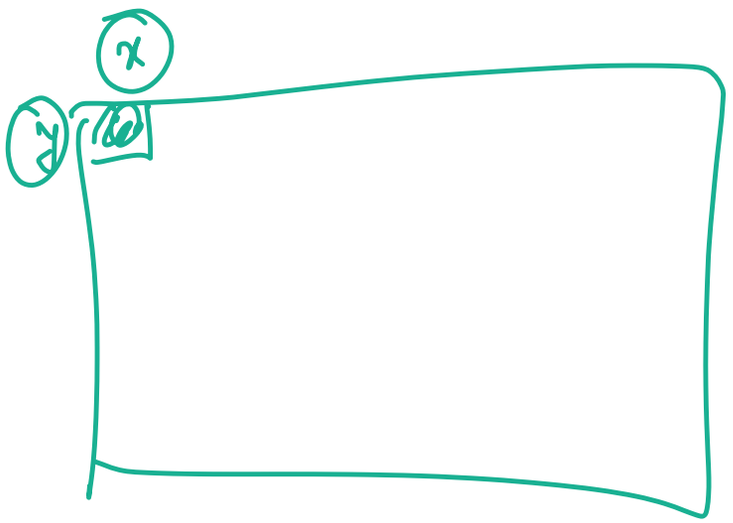
7 x 7 x 30

label vector.

2

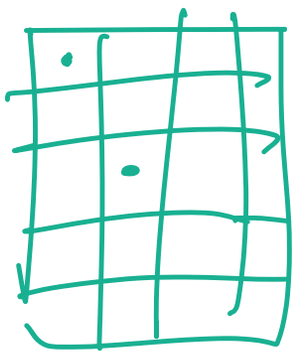
→ zero.

$$R(\theta) = \begin{bmatrix} \cos\theta & \sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$



$$R(\theta) \cdot \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} x_{new} \\ y_{new} \end{bmatrix}$$



$q = k$

$q \times w \times h = 300$

$r00$

$w \times h = \frac{\cancel{300}}{q\ 3}$