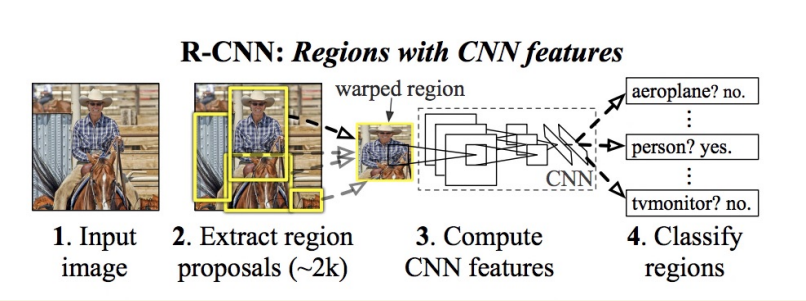
Object Detection

Object detection algorithm

R-CNN:

extract 2000 regions from raw image.

region proposals (2000 region proposals generated by selective search algorithm)

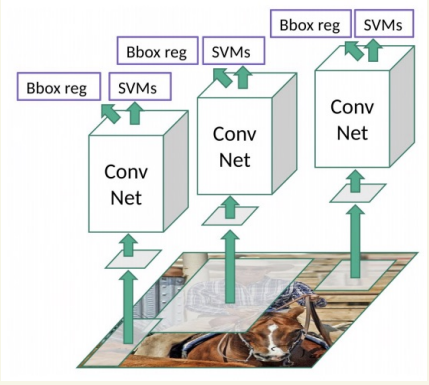


↳ 2000 candidate region proposals warped into a square, fed into a CNN.

CNN produces a 4096-dim feature vector as output.

Features feed into SVM to classify

Predict 4 offset values of bounding box.



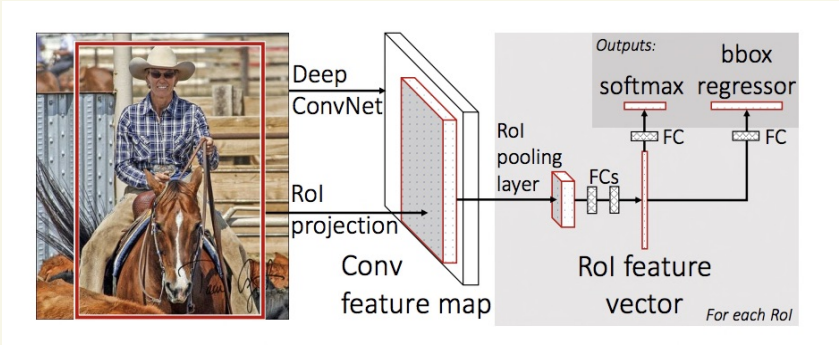
problems with R-CNN:

① waste of time to train (classify 2000 region per img)

② Not suitable for Real-time (around 47s per img)

③ Selective search algorithm is fixed. (could generate bad region proposals)

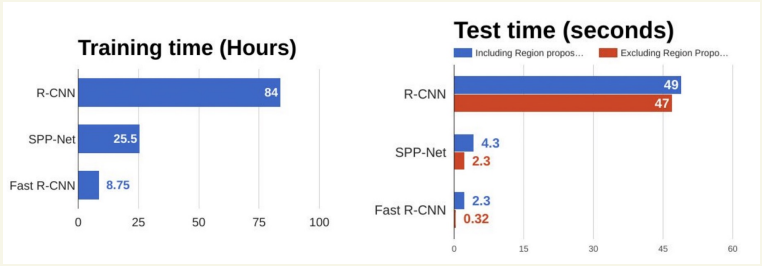
Fast R-CNN → solve drawbacks of R-CNN.



(R-CNN) feed region proposals to CNN

(Fast R-CNN) feed input img to CNN → (generate) Convolutial feature map

→ { identify the region of proposals; wrap them into squares } → (reshape into fixed size (RoI pooling layer)) → feed into FC layer → { softmax; bbox reg }



No need to feed 2000 region proposals to CNN

Convolution operation Once per img.

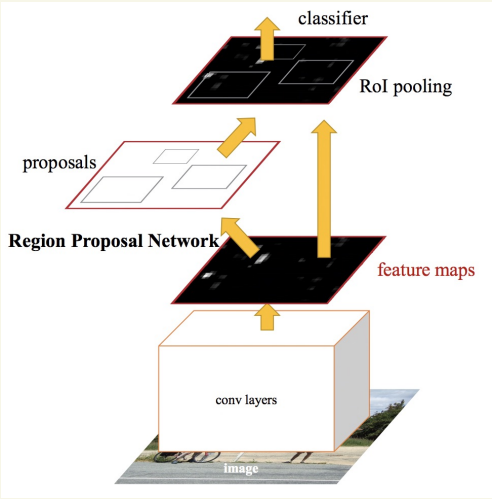
region proposals : bottlenecks in Fast R-CNN

Faster R-CNN

slow, time-consuming

R-CNN & Fast R-CNN : Selective search

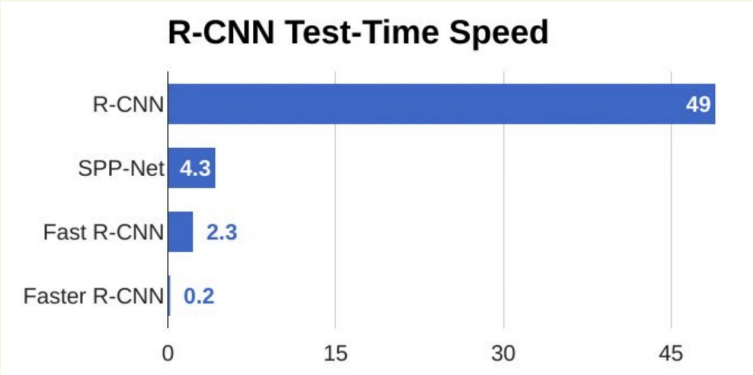
Faster R-CNN : "eliminated"



img —CNN→ convolutional feature map

~~selective search~~ a separate network

predict the region proposals —RoI pooling layer→ reshape to fixed size → { classify / predict offset values



Much faster than before

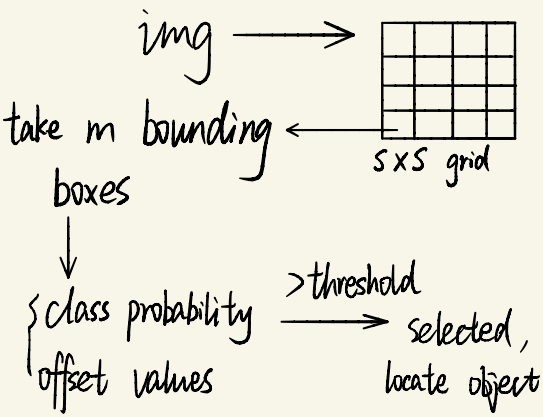
suitable for real-time object-detection

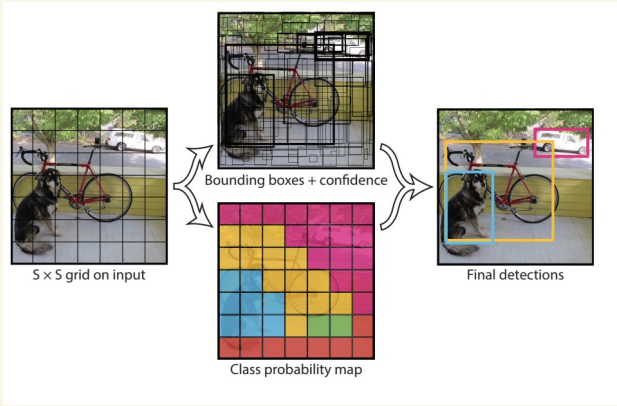
YOLO

YOLO — You Only Look Once

R-CNN, Fast R-CNN, Faster R-CNN : use regions to localize object.

YOLO : a single CNN predicts { bounding boxes → class probabilities for boxes





far more faster (45 fps)

struggle with small objects.