Object Detection

Sihao Liang
Jiajun Lu
Kevin Perkins

Overview

Intro

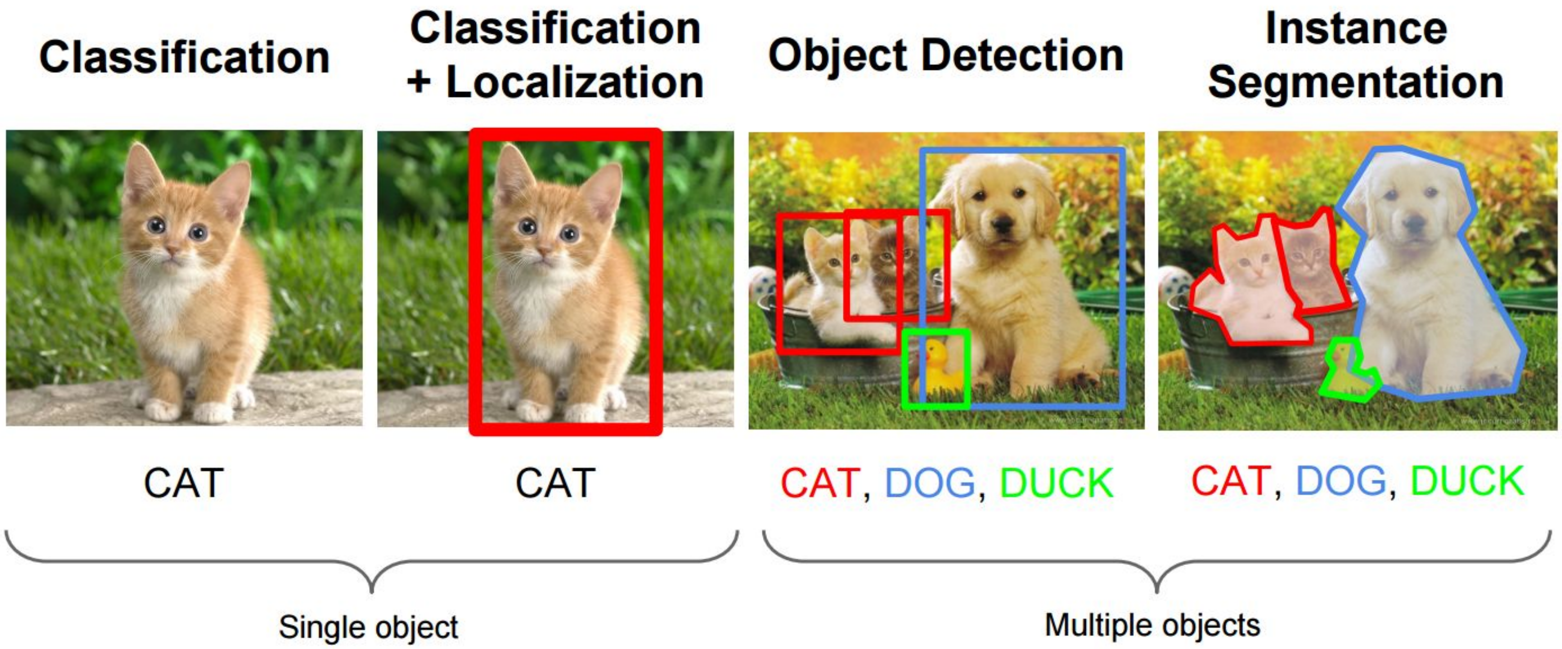
Part I: Two Stage Detection

Part II: Unified Detection

Part III: Others

Summary and comparison

What is object detection



http://cs231n.stanford.edu/slides/winter1516_lecture8.pdf

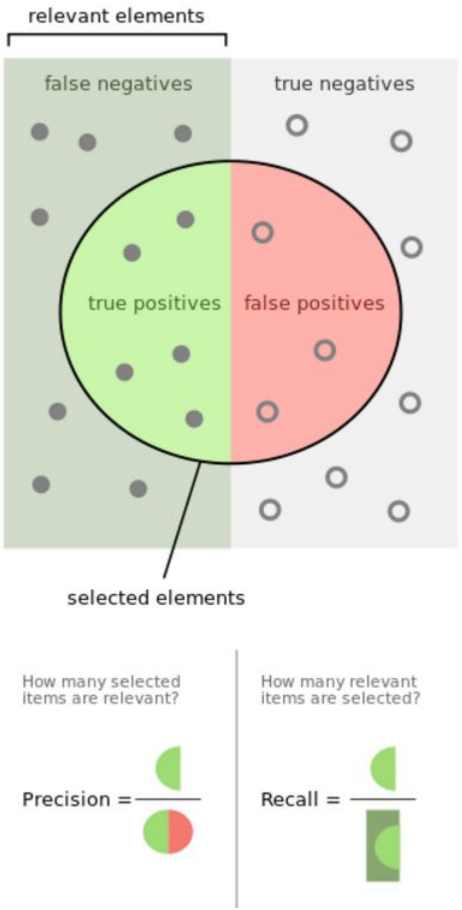
Terms

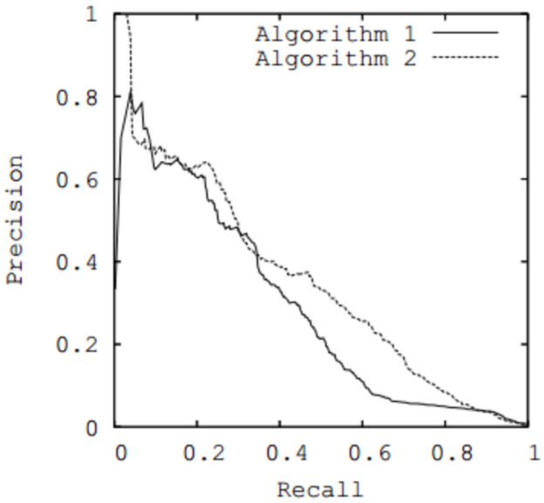
Recall

Precision

mAP

IoU





https://en.wikipedia.org/wiki/Precision_and_recall

Detection Competitions

Pascal VOC

COCO

ImageNet ILSVRC

VOC: 20 classes

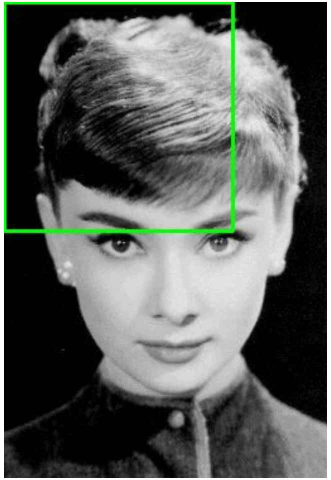
COCO: 200 classes

http://host.robots.ox.ac.uk/pascal/VOC/voc2012/index.html#introduction

Before Deep Learning

Sliding windows.

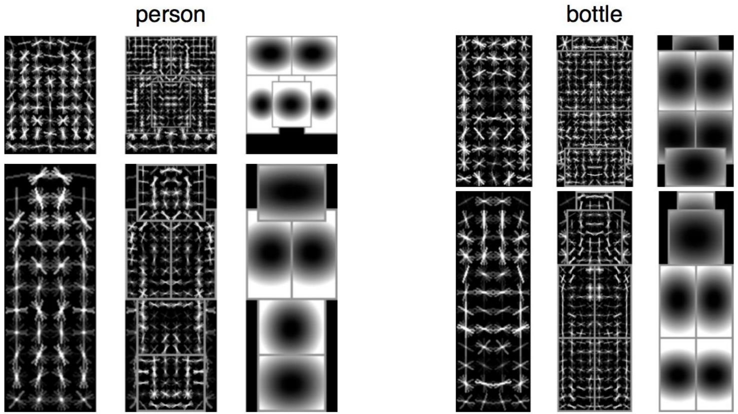
- Score every subwindow.



http://www.pyimagesearch.com/2014/11/10/histogram-oriented-gradients-object-detection/

Deformable part models (DPM)

- Uses HOG features
- Very fast



https://cs.brown.edu/~pff/papers/lsvm-pami.pdf

Selective Search

http://www.huppelen.nl/publications/selectiveSearchDraft.pdf

Hard Negative Mining

Imbalance between positive and negative examples.

Use negative examples with higher confidence score.

Non Maximum Suppression

If output boxes overlap, only consider the most confident.

Bounding Box Regression

Regression used to find bounding box parameters

Applied in one of two ways

- Bounding box refinement
- Complete object detection

Example Loss Function

$$\sum_{i\ in\ I} [(\boxed{x_i} - \boxed{\hat{x}_i})^2 + (\boxed{y_i} - \boxed{\hat{y}_i})^2 + (\boxed{\sqrt{w_i}} - \boxed{\sqrt{\hat{w}_i}})^2 + (\boxed{\sqrt{h_i}} - \boxed{\sqrt{\hat{h}_i}})^2]$$

I is the set of all matching bounding boxes (Highest IoU with ground truth)

Two Stage Detection

Part I

RCNN : Region Proposal + CNN

Use selective search to come up with regional proposal

First object detection method using CNN

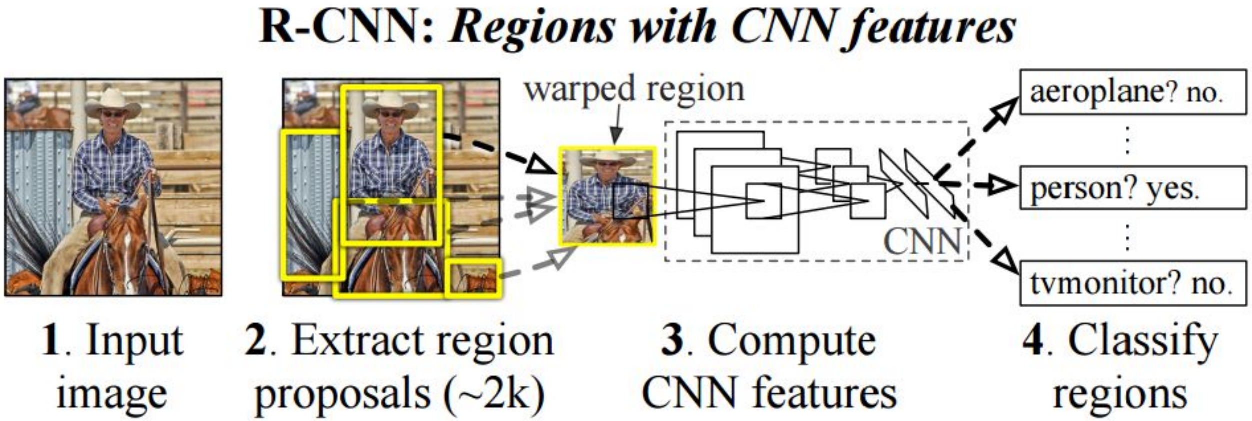
Rich feature hierarchies for accurate object detection and semantic segmentation

Ross Girshick Jeff Donahue Trevor Darrell Jitendra Malik

Nov 2013

https://people.eecs.berkeley.edu/~rbg/papers/r-cnn-cvpr.pdf

RCNN :



https://people.eecs.berkeley.edu/~rbg/papers/r-cnn-cvpr.pdf

Training RCNN

Step1: train your own CNN model for classification (or use existing model), using ImageNet dataset.

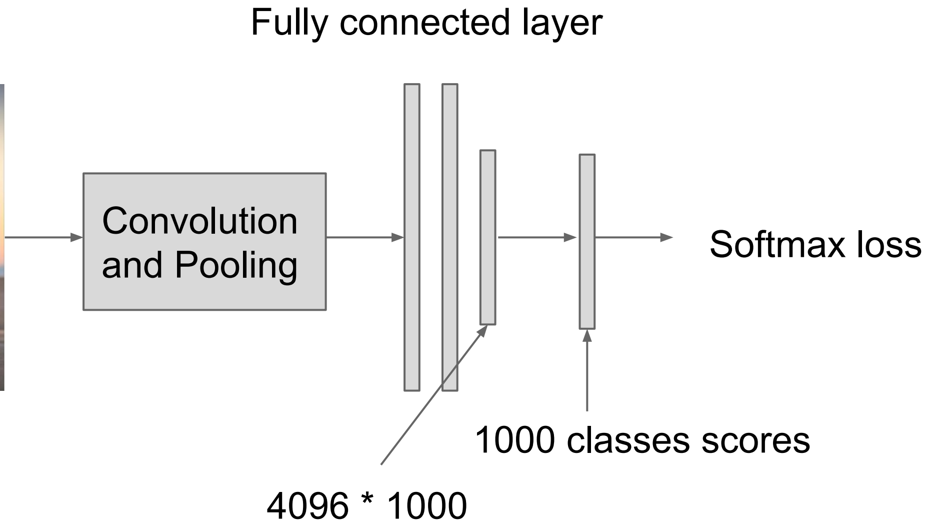
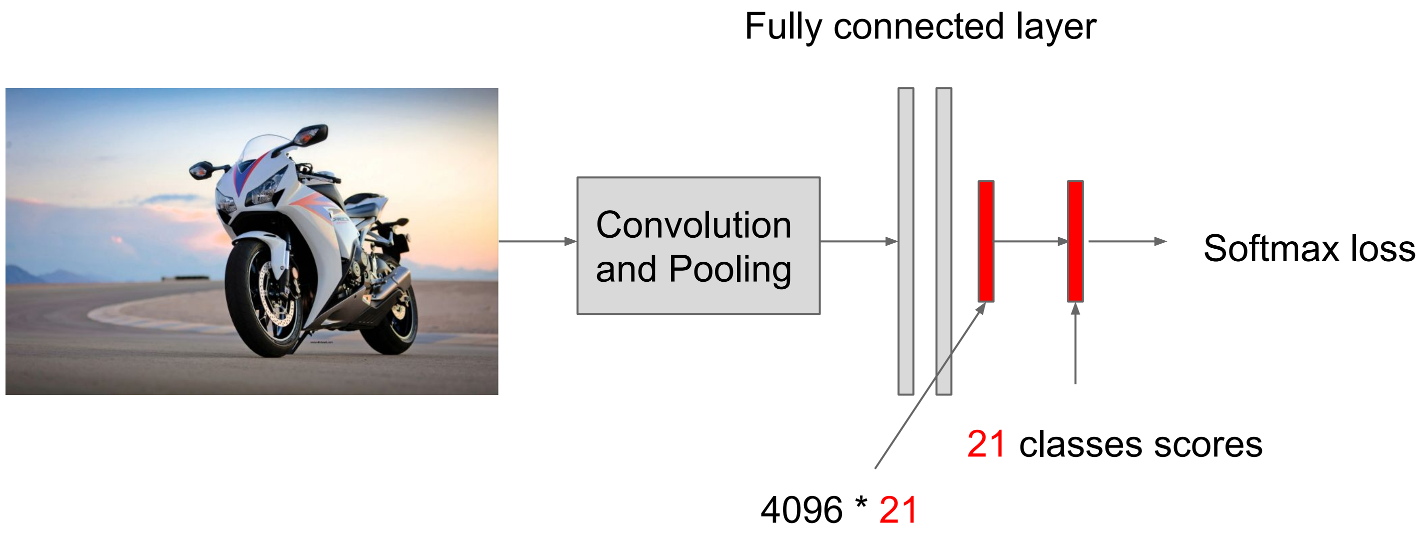


Image source : http://www.shunvmall.com/bike-pic/47539009.html

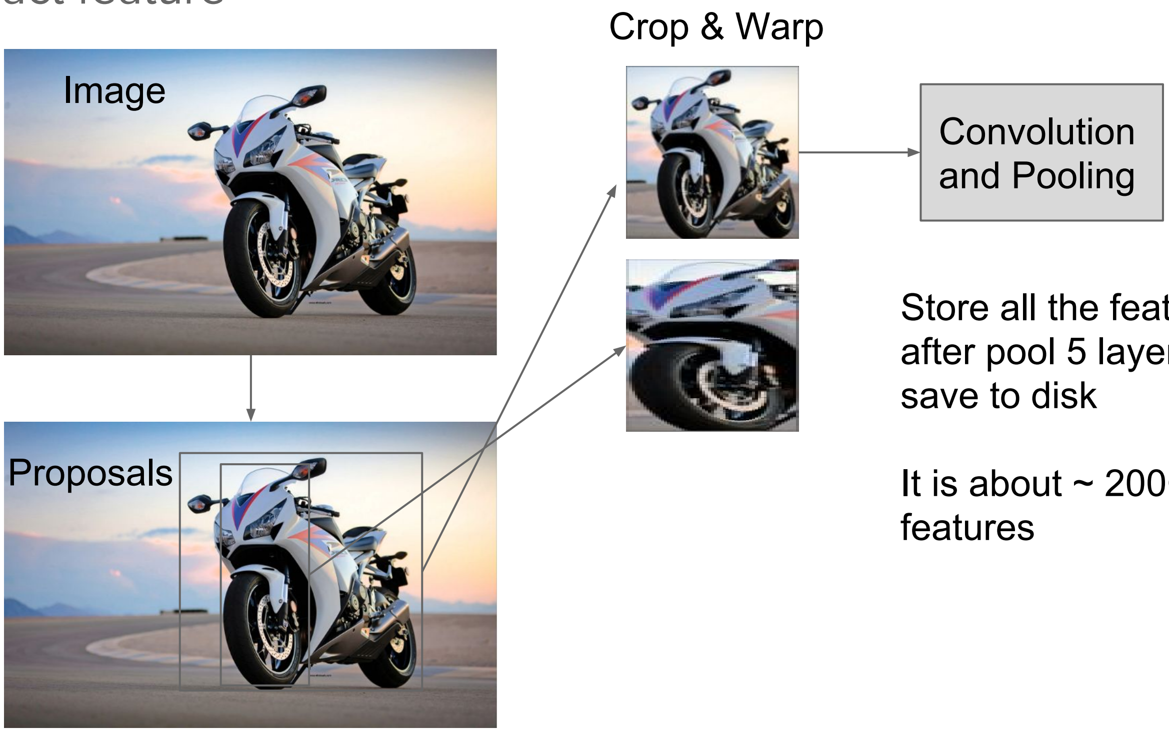
Training RCNN

Step2: focus on 20 classes + 1 background. Remove the last FC layer and replace it with a smaller layer and fine-tune the model using PASCAL VOC dataset



Training RCNN

Step3: extract feature



Store all the features after pool 5 layer and save to disk

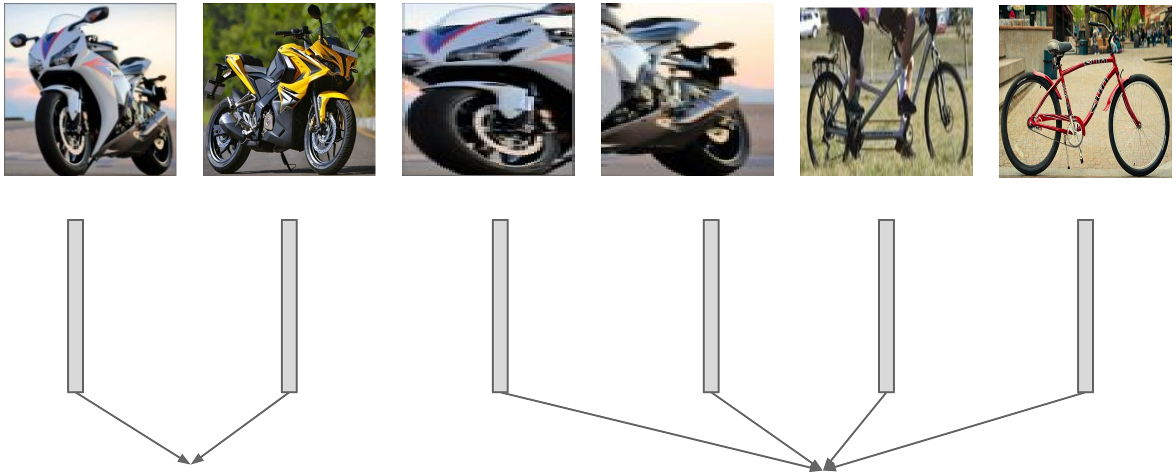
It is about ~ 200G features

Training RCNN

Step4: train SVM for each class

Crop / Warp image

Features from last step

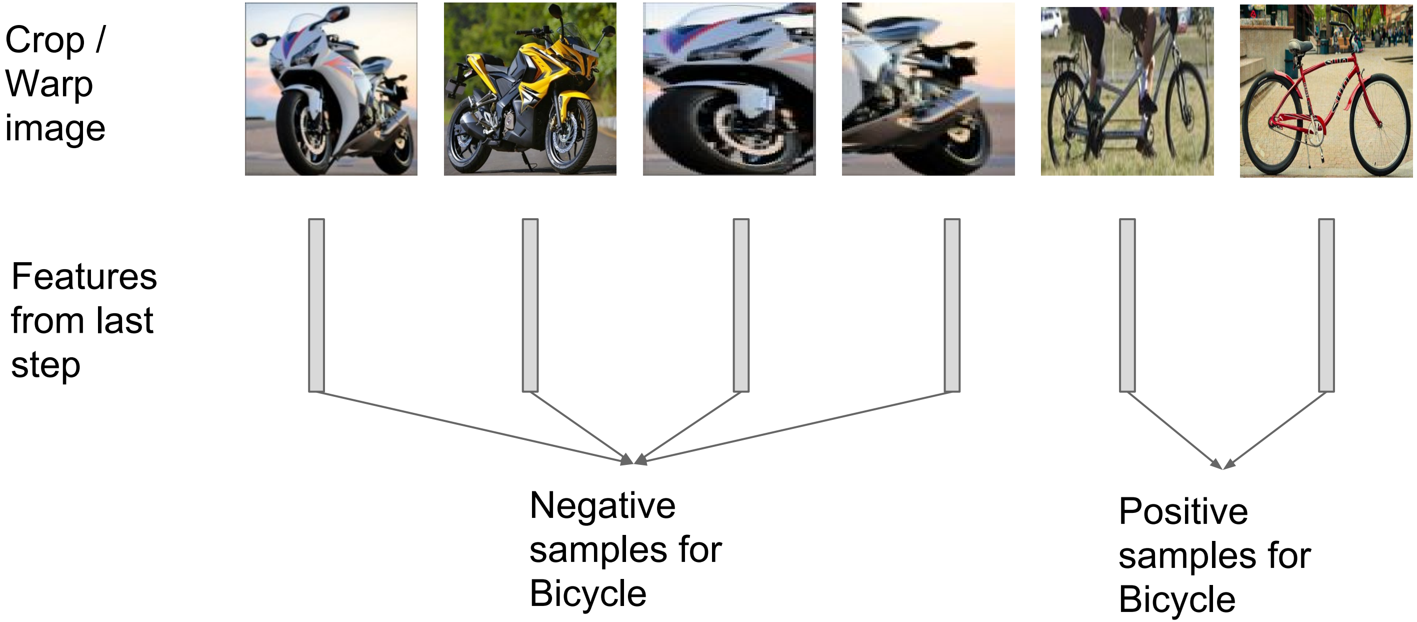


Positive samples for Motorbike

Negative samples for Motorbike

Training RCNN

Step4: train SVM for each class



Fast RCNN

Share convolution layers for proposals from the same image

Faster and More accurate than RCNN

ROI Pooling

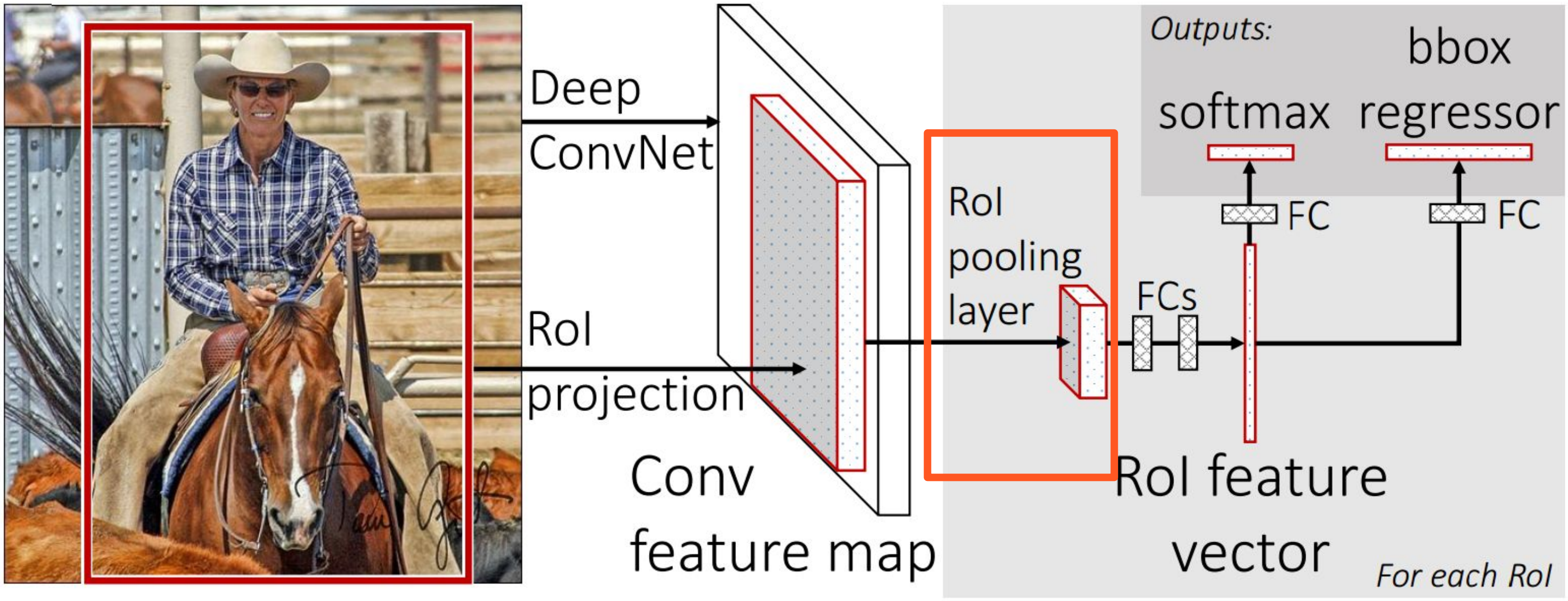
Fast R-CNN

Ross Girshick

Apr 2015

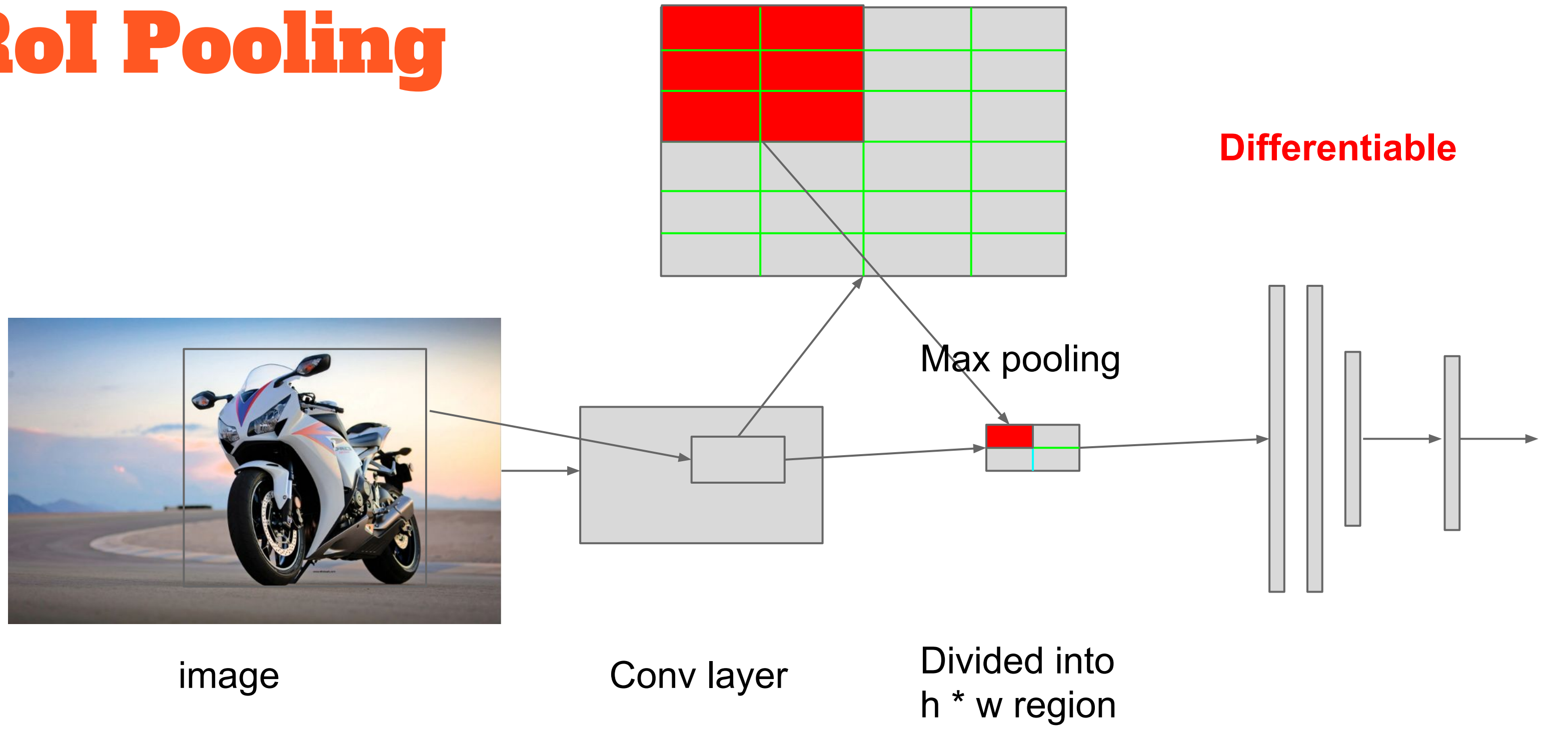
https://arxiv.org/pdf/1504.08083v2.pdf

Fast RCNN



https://arxiv.org/pdf/1504.08083v2.pdf

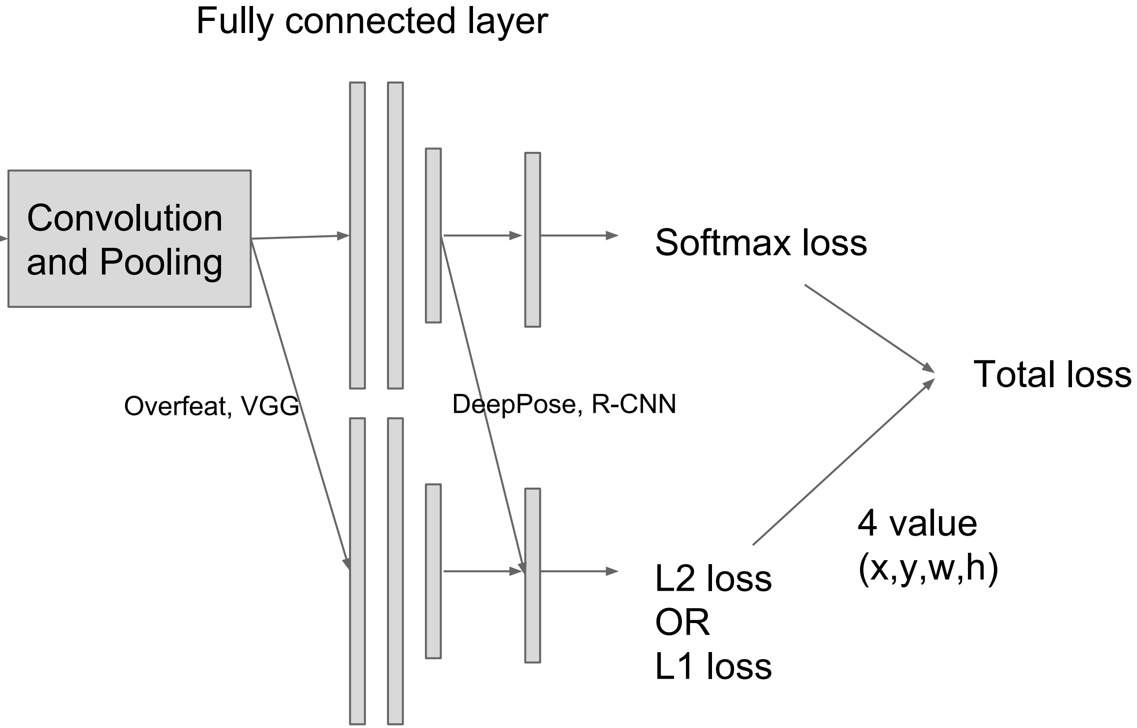
RoI Pooling

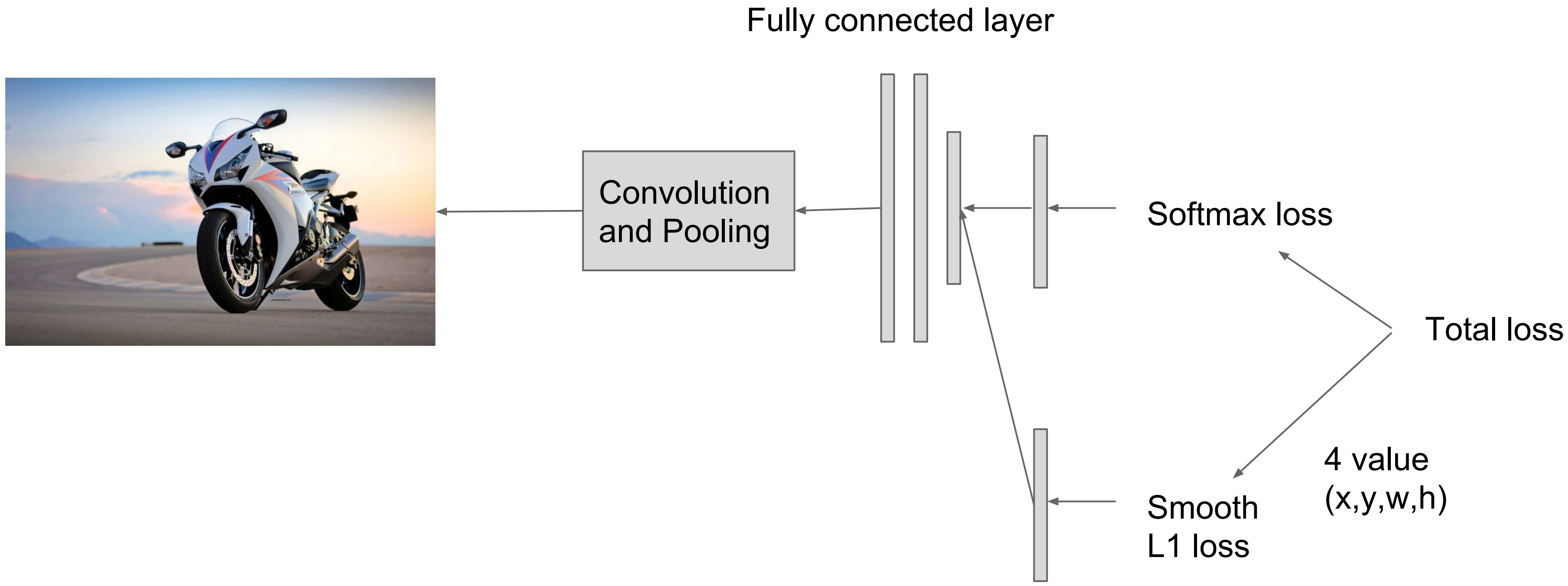


Differentiable
Max pooling
image
Conv layer
Divided into
h * w region

What is bbox-regressor?

Bounding box regression




Fully connected layer
Convolution and Pooling
Softmax loss
Total loss
4 value (x,y,w,h)
Smooth L1 loss

Result compare

	Fast R-CNN	R-CNN
Train time (h)	9.5	84
-speedup	8.8x	1x
Test time/image	0.32s	47.00 s
-test speedup	146x	1x
mAP	66.9	66

Trained using VGG 16 on Pascal VOC 2007 dataset
Not including proposal time

Source: R. Girshick

Result compare

	Fast R-CNN	R-CNN
Test time/image	0.32s	47.00 s
-test speedup	146x	1x
Test time/image with proposal	2s	50 s
-test speedup	25x	1x

Source: cs231 standford

Faster RCNN

Don’t need to have external regional proposals

RPN - Regional Proposal Network

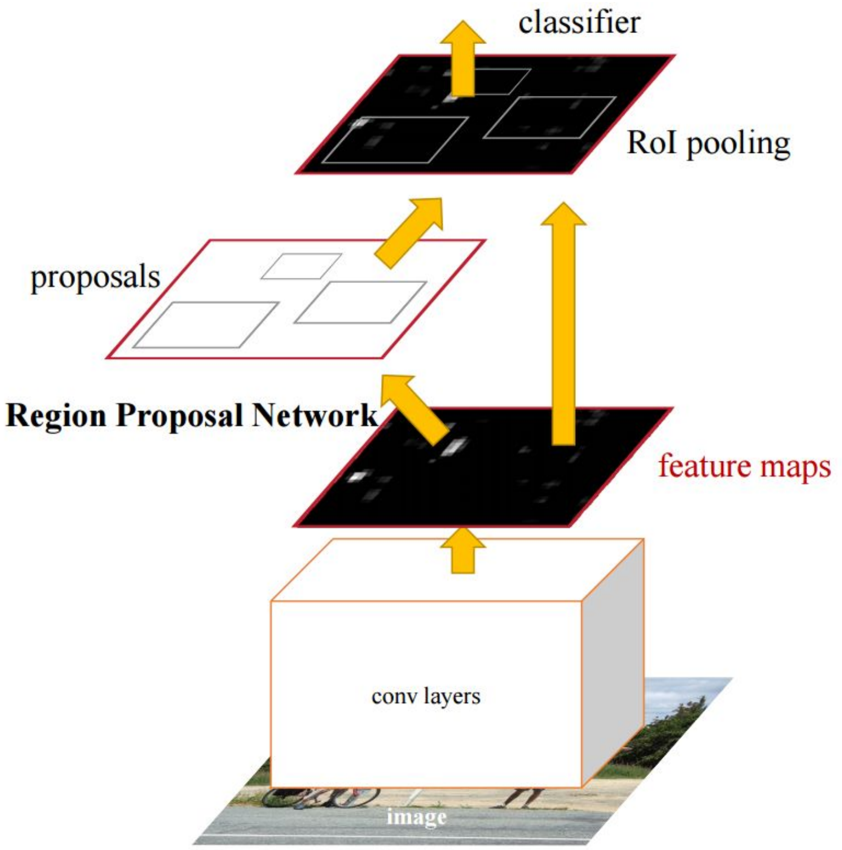
Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun

Jun 2015

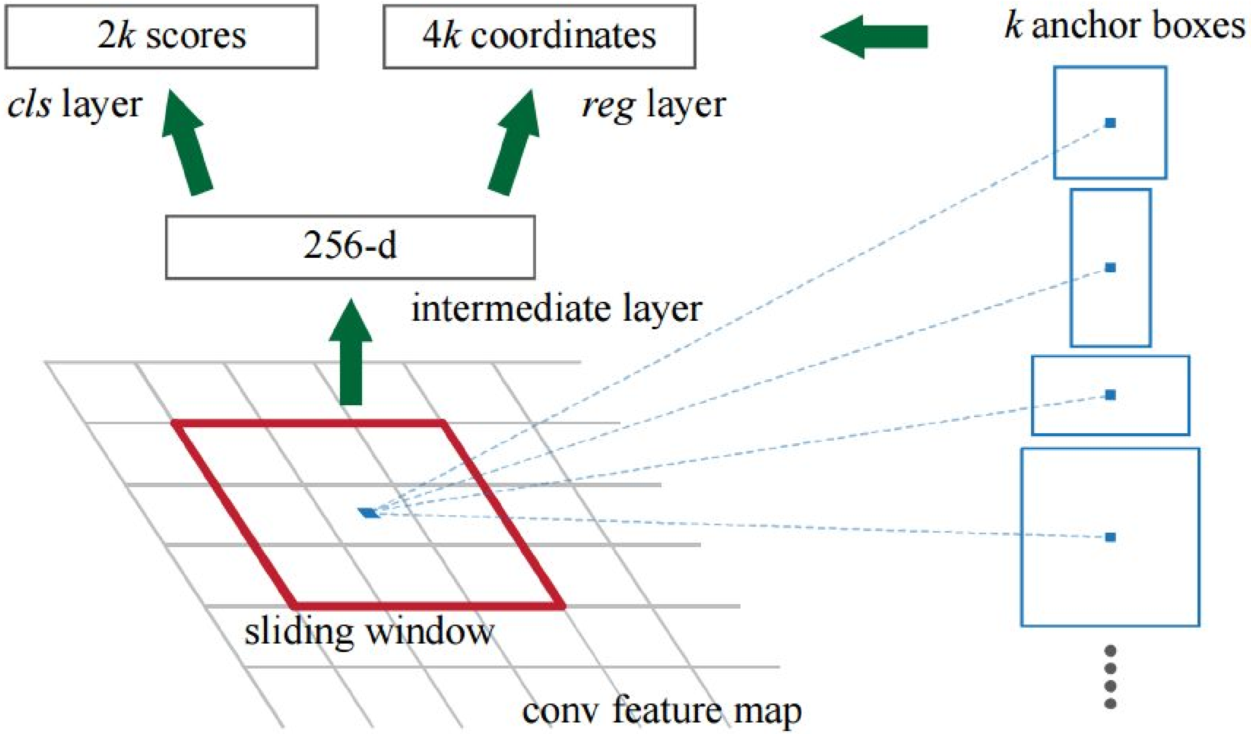
https://arxiv.org/pdf/1506.01497v3.pdf

Faster RCNN



https://arxiv.org/pdf/1506.01497v3.pdf

Faster RCNN



https://arxiv.org/pdf/1506.01497v3.pdf

Result compare

	Faster R-CNN	Fast R-CNN	R-CNN
Test time/image With proposal	0.2S	2s	50s
-test speedup	250x	25x	1x
mAP	66.9	66.9	66

Trained using Pascal VOC 2007 dataset

Source: cs231 standford

R-FCN :Region-based Fully Convolutional Networks

Use position sensitive score map

Share all conv and fc layers between all proposals for the same image

R-FCN: Object Detection via Region-based Fully Convolutional Networks

Jifeng Dai, Yi Li, Kaiming He, Jian Sun

May 2016

https://arxiv.org/pdf/1605.06409v2.pdf

R-FCN

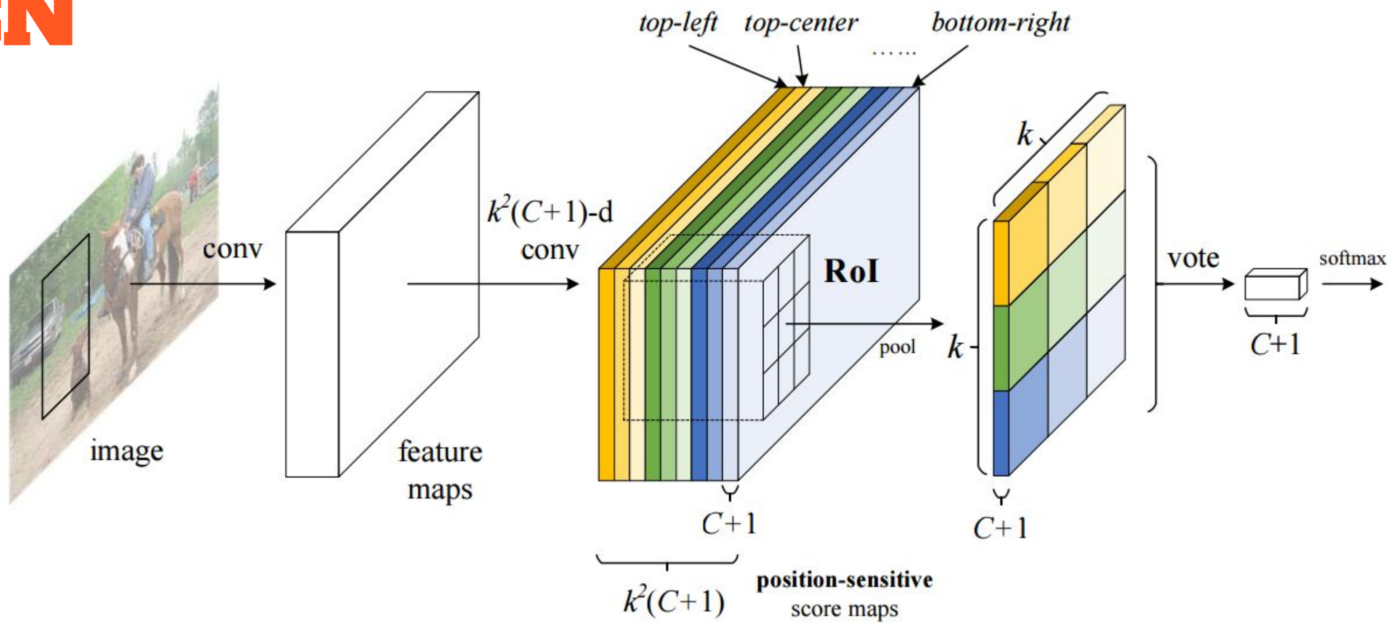


Figure 1: Key idea of **R-FCN** for object detection. In this illustration, there are $k \times k = 3 \times 3$ position-sensitive score maps generated by a fully convolutional network. For each of the $k \times k$ bins in an RoI, pooling is only performed on one of the k^2 maps (marked by different colors).

https://arxiv.org/pdf/1605.06409v2.pdf

R-FCN

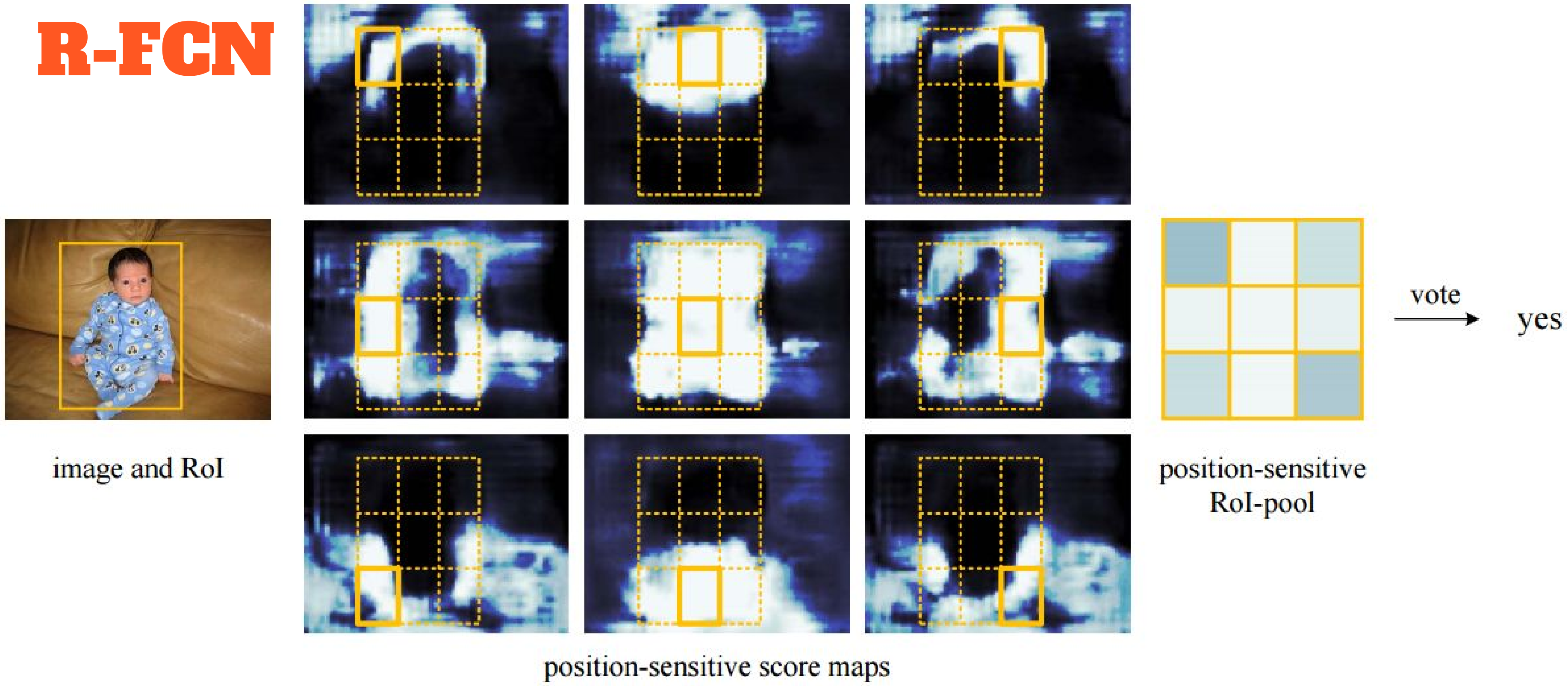


Figure 3: Visualization of R-FCN ($k \times k = 3 \times 3$) for the *person* category.

https://arxiv.org/pdf/1605.06409v2.pdf

R-FCN

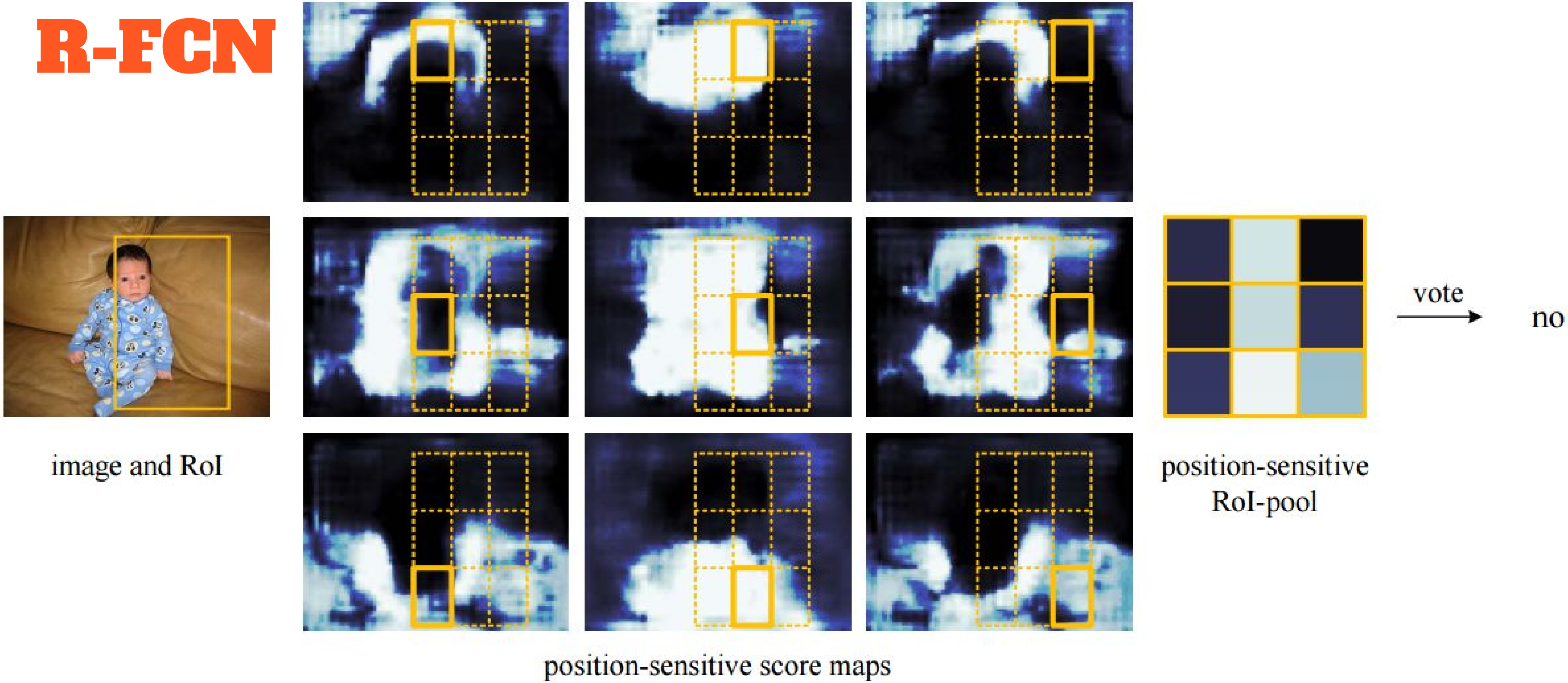


Figure 4: Visualization when an RoI does not correctly overlap the object.

https://arxiv.org/pdf/1605.06409v2.pdf

Methodologies Compare

	RCNN	Faster RCNN	RFCN
Depth of shared convolutional subnetwork	0	91	101
Depth of ROI-wise subnetwork	101	10	0

Trained using ResNet 101

Result compare

	Faster R-CNN	R-FCN
Test time/image With proposal	0.42S	0.2s
mAP	76.4	76.6

Trained using ResNet 101 on Pascal VOC 2007 dataset

Unified Detection

Part II

Problems with 2 step detection.

Complex Pipeline

Slow (Cannot run in real time)

Hard to optimize each component

Yolo: You Only Look Once

Consider detection a regression problem

Use a single ConvNet

Runs once on entire image. Very Fast!

You only look once: Unified, real-time object detection.
Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi
June 2015

How it works

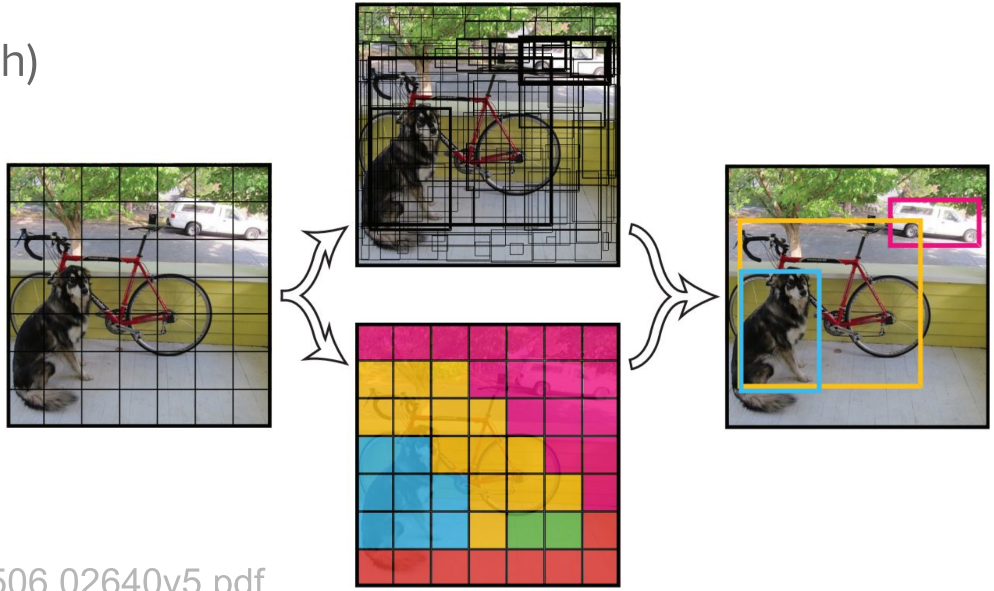
The following predictions are made for each cell in an S x S grid.

C conditional class probabilities **Pr(Class$_i$ | Obj)**

B bounding boxes (4 parameters each)

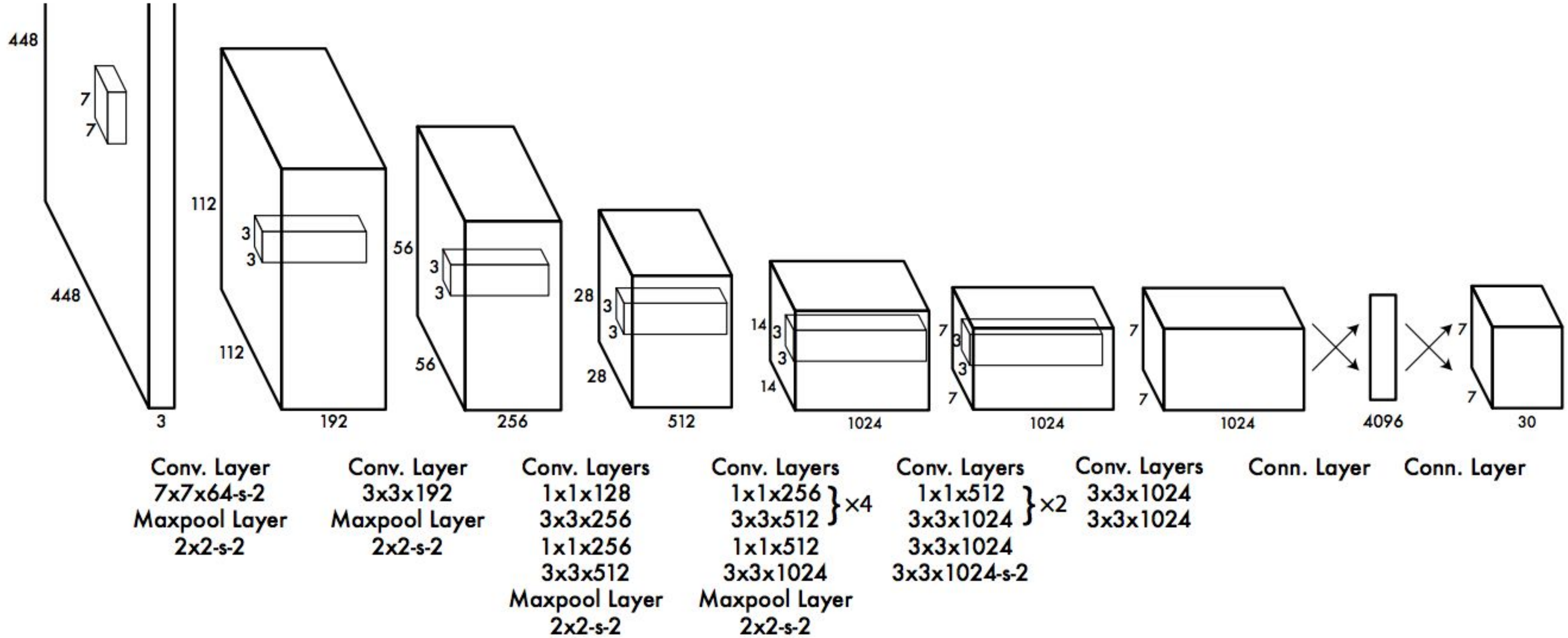
B confidence scores **Pr(Obj)*IoU**

Output is **S x S x (5B+C)** tensor



https://arxiv.org/pdf/1506.02640v5.pdf

Architecture



https://arxiv.org/pdf/1506.02640v5.pdf

Performance (VOC 2007)

	Yolo	Faster R-CNN (VGG-16)
mAP	63.4	73.2
FPS	45	7

Trained on Pascal VOC 2007 + 2012 dataset

https://arxiv.org/pdf/1506.02640v5.pdf

Limitations of Yolo

Struggles with small objects

Struggles with unusual aspect ratios

Poor localization

SSD Single Shot Detector

Faster than Yolo, as accurate as Faster R-CNN

Predicts categories and box offsets

Uses small convolutional filters applied to feature maps

Makes predictions using feature maps of different scales

SSD: Single shot multibox detector

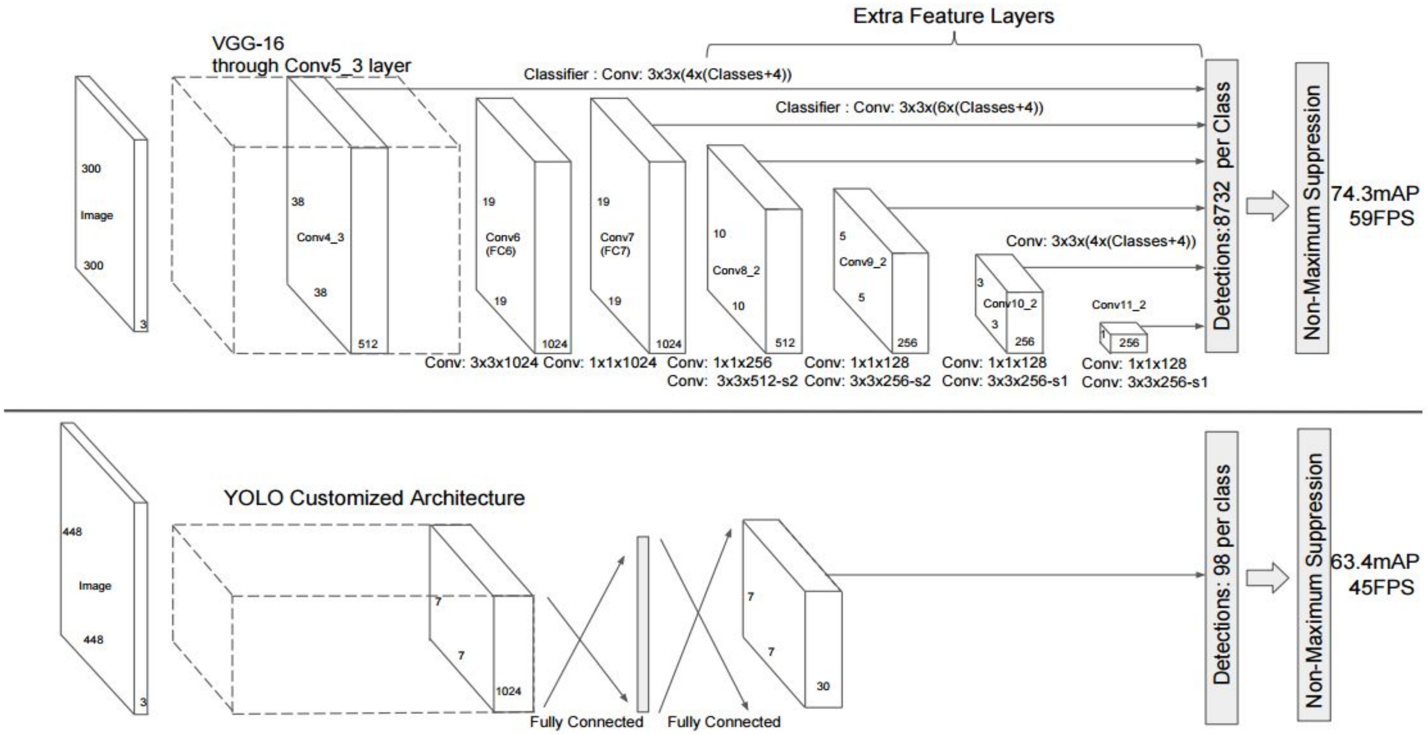
Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, Alexander C. Berg

Dec 2015

Comparison to Yolo

SSD

Yolo



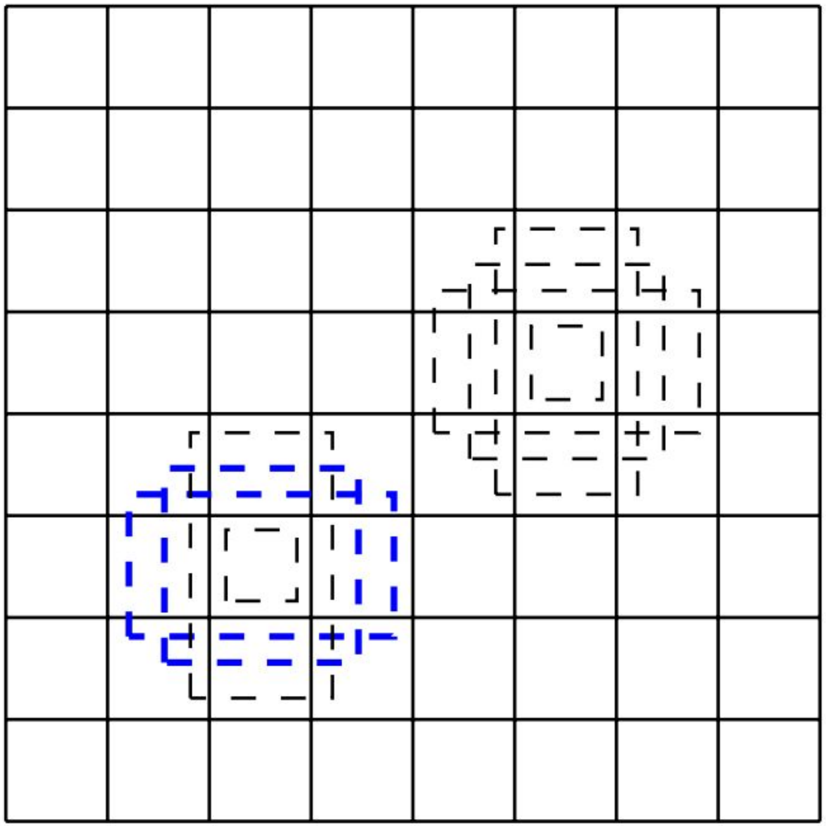
https://arxiv.org/pdf/1512.02325v5.pdf

Default Boxes

Multiple aspect ratios per cell.

Similar to Faster R-CNN Anchor Boxes.

- Applied to many feature maps.



https://arxiv.org/pdf/1512.02325v5.pdf

SSD Detector

Detectors are convolutional filters.

Each detector outputs a single value.

Predict class probabilities.

Predict bounding box offsets.

(classes + 4) detectors are needed for a detection.

(classes + 4) x (#default boxes) x m x n outputs for a mxn feature map.

Results (VOC 2007)

	SSD*	Yolo*	Faster R-CNN*
mAP	74.3	66.4	73.2
FPS	46	21	7

*VGG16

Trained on Pascal VOC 2007 + 2012 dataset

https://arxiv.org/pdf/1512.02325v5.pdf

Others

Part III

Feature Pyramid Networks

Uses ConvNet as Feature Pyramid

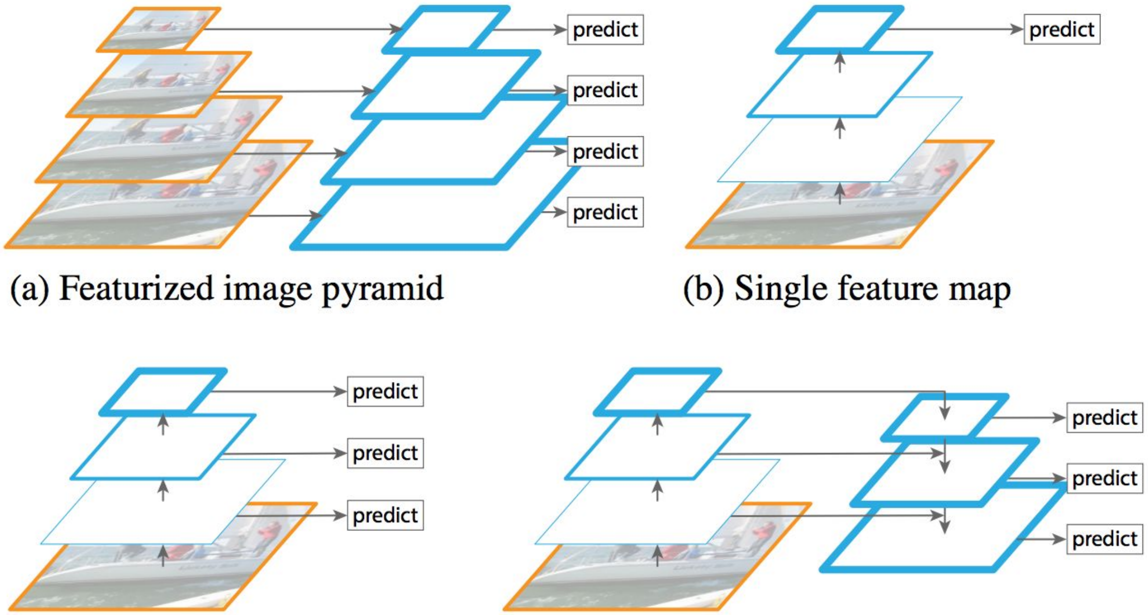
Includes low level feature maps to detect small objects

Top down pathway provides contextual information

Feature Pyramid Networks for Object Detection.
Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, Serge Belongie.
Dec 2016

Comparison to prior methods

a. Accurate but slow.
b. Misses low level information. (Yolo)
c. Misses context in low level predictions. (SSD)
d. Accurate and fast.



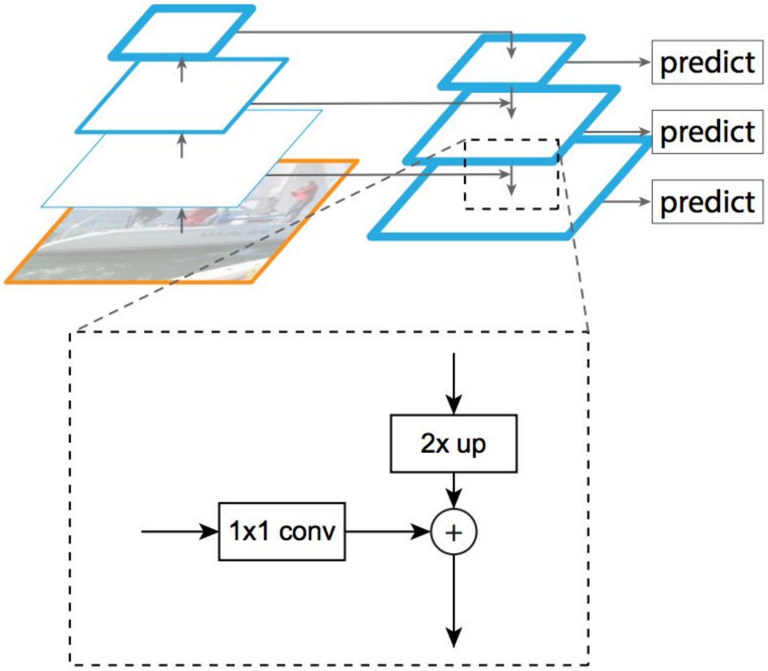
(a) Featurized image pyramid (b) Single feature map

(c) Pyramidal feature hierarchy (d) Feature Pyramid Network

https://arxiv.org/pdf/1512.02325v5.pdf

Top down pathway

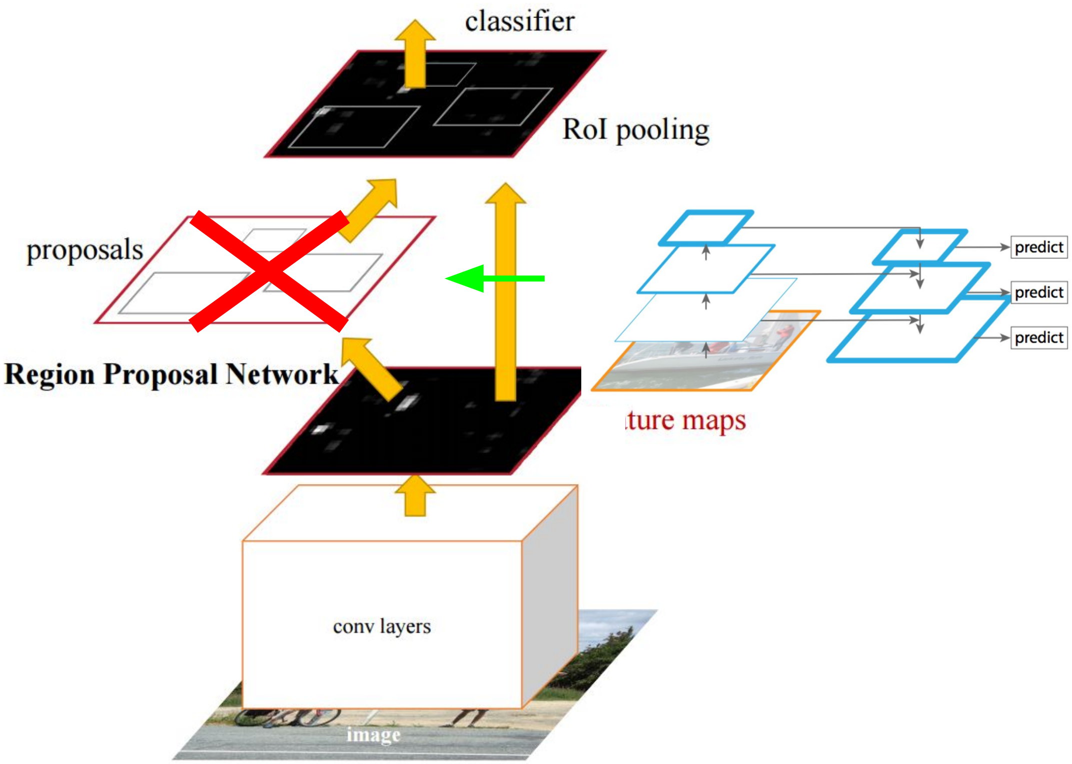
- High level (semantically strong) feature maps are upsampled.
- Lateral connections merge feature maps from bottom up pathway.



https://arxiv.org/pdf/1512.02325v5.pdf

Feature Pyramid Networks for RPN

- Replace single scale feature map with FPN.
- Single scale anchors at each level.



https://arxiv.org/pdf/1512.02325v5.pdf

Results

- COCO
- State of the art single-model

	Faster R-CNN on **FPN***	Faster R-CNN +++*	ION**
mAP	36.2	34.9	31.2
mAP (small images)	18.2	15.6	12.8

* ResNet-101
** VGG-16

https://arxiv.org/pdf/1512.02325v5.pdf

ION : Inside-Outside Network

Use multi-scale Conv features for inside region

Use four direction RNN features for outside region

Inside-Outside Net: Detecting Objects in Context with Skip Pooling and Recurrent Neural Networks

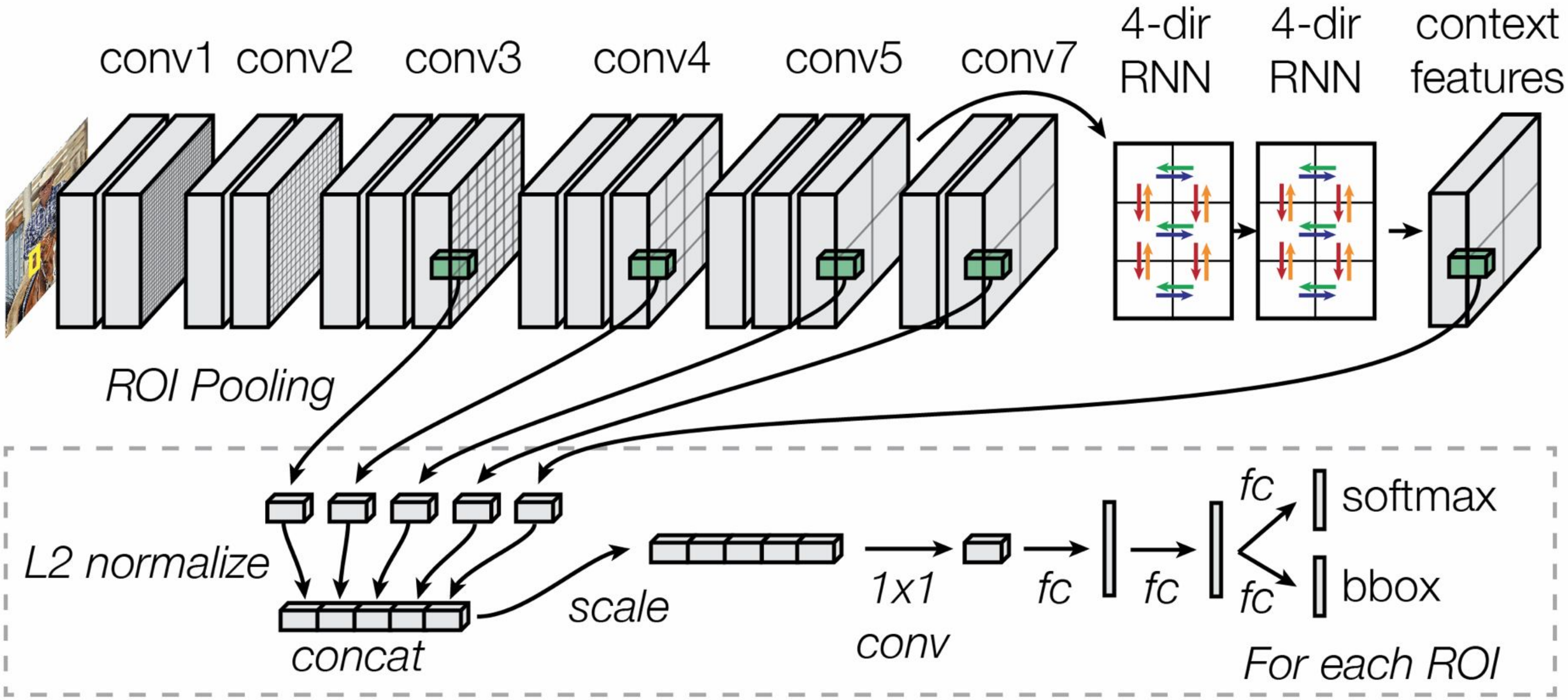
Sean Bell, C.Lawrence Zitnick, Kavita Bala, Ross Girshick

Cornell University, Microsoft Research

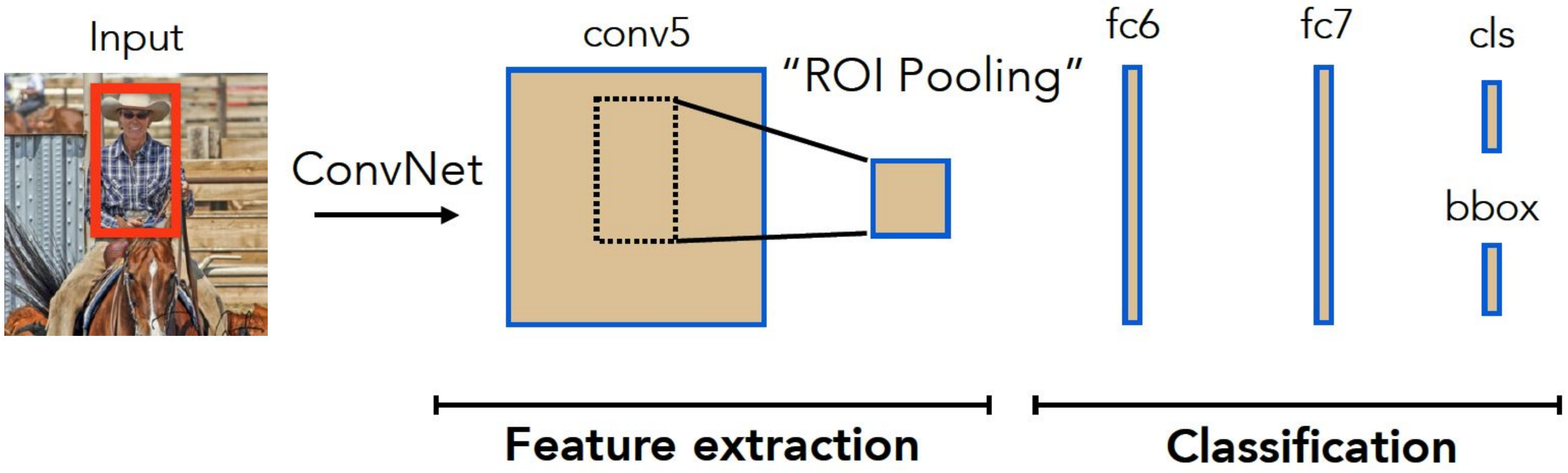
CVPR 2016

https://people.eecs.berkeley.edu/~rbg/papers/r-cnn-cvpr.pdf

ION: Inside-Outside Net



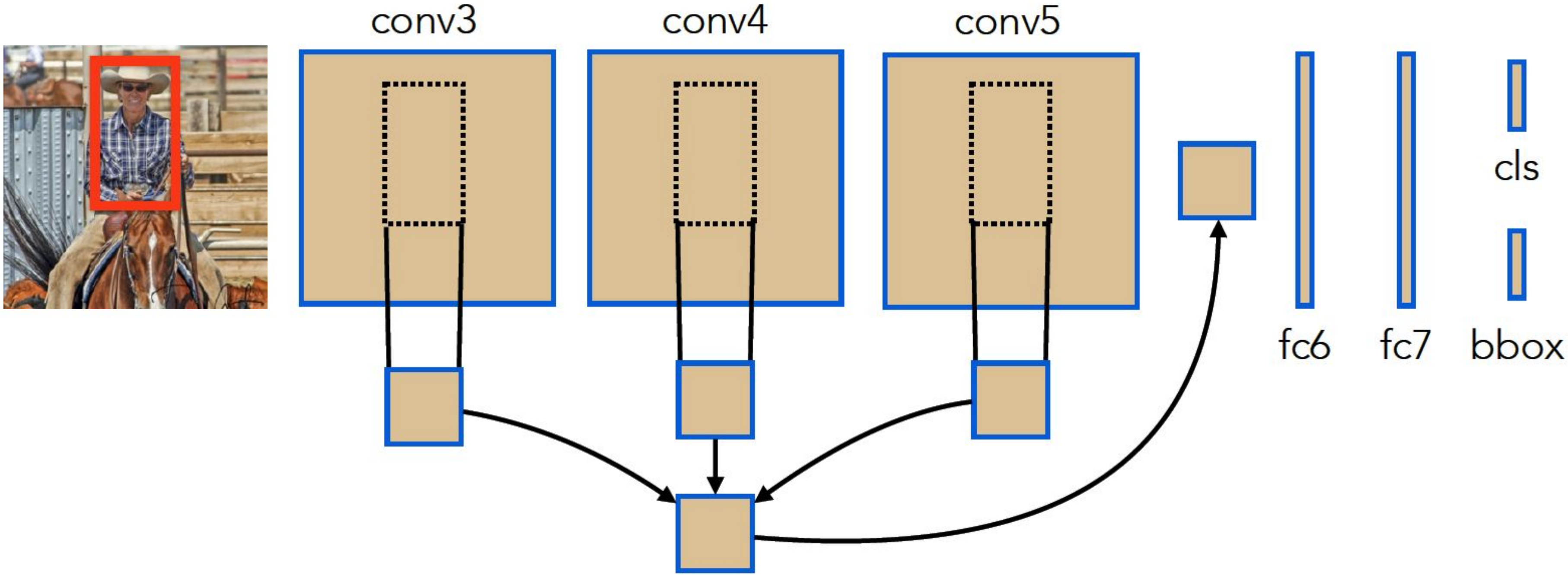
https://www.robots.ox.ac.uk/~vgg/rg/slides/ion-coco.pdf



Can we improve on feature extraction?

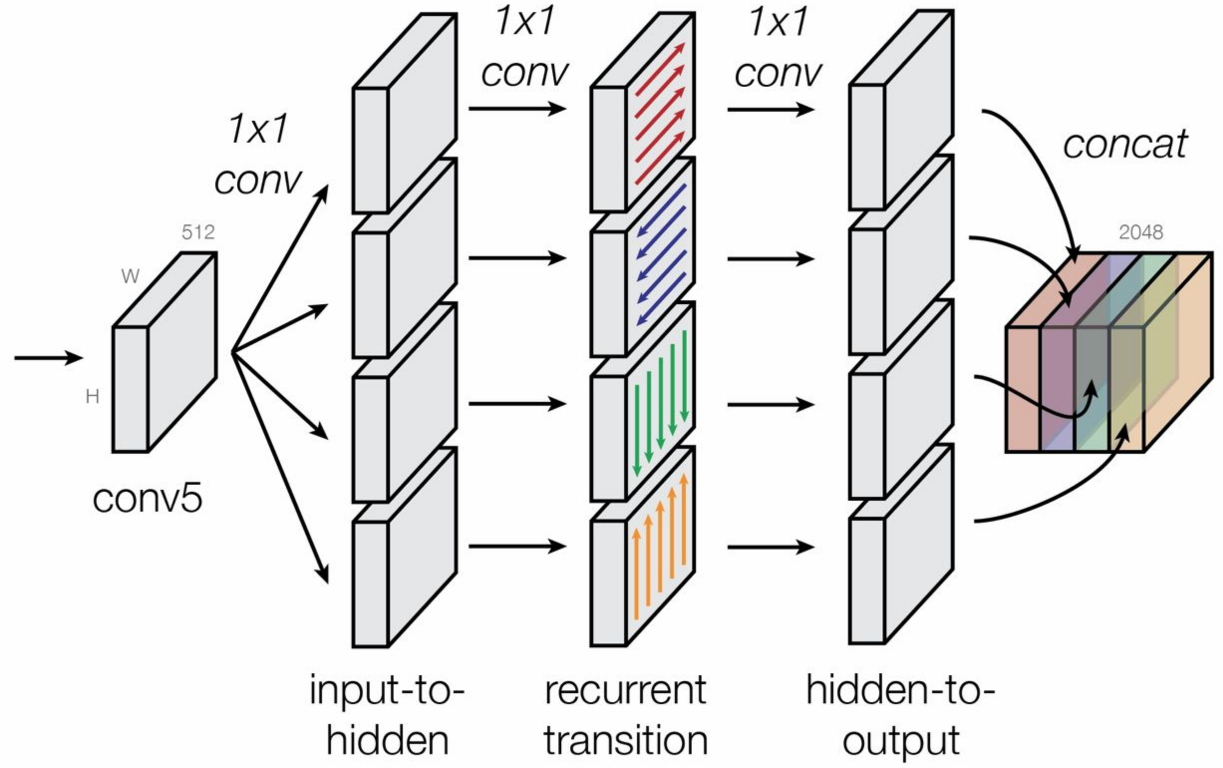
- For small objects, the footprint on conv5 might only cover a 1x1 cell, which gets upsampled to 7x7
- Only local features (inside the ROI) are used for classification

COMBINING ACROSS LAYERS

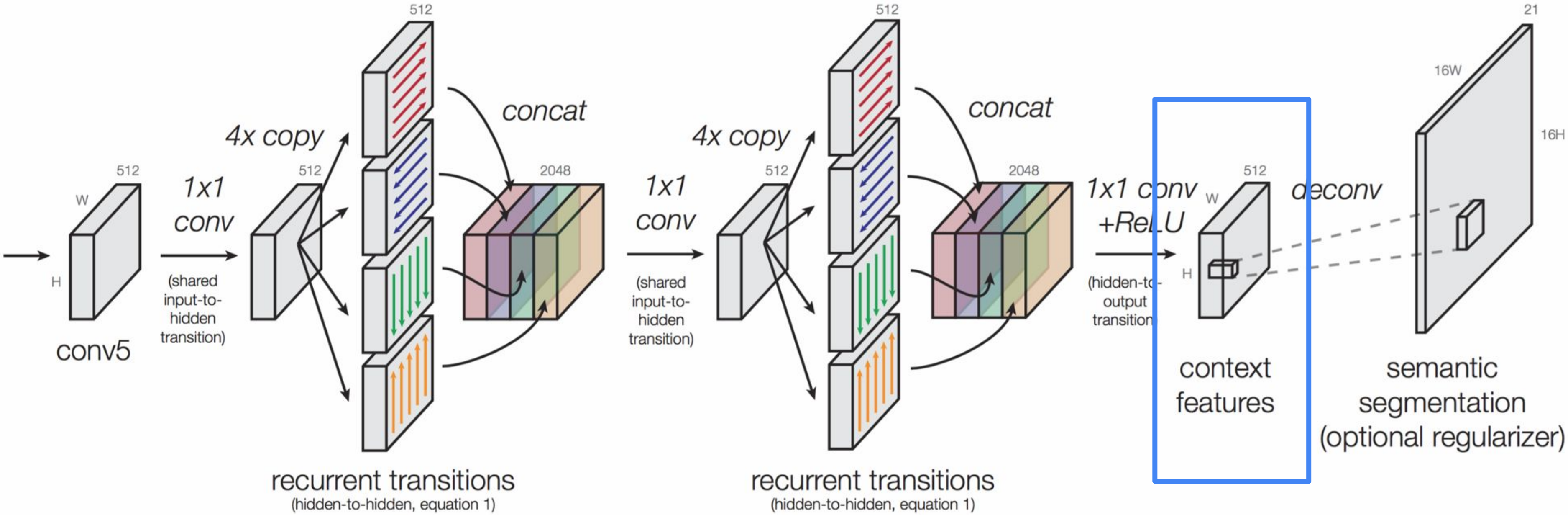


normalize, concatenate, **re-scale**

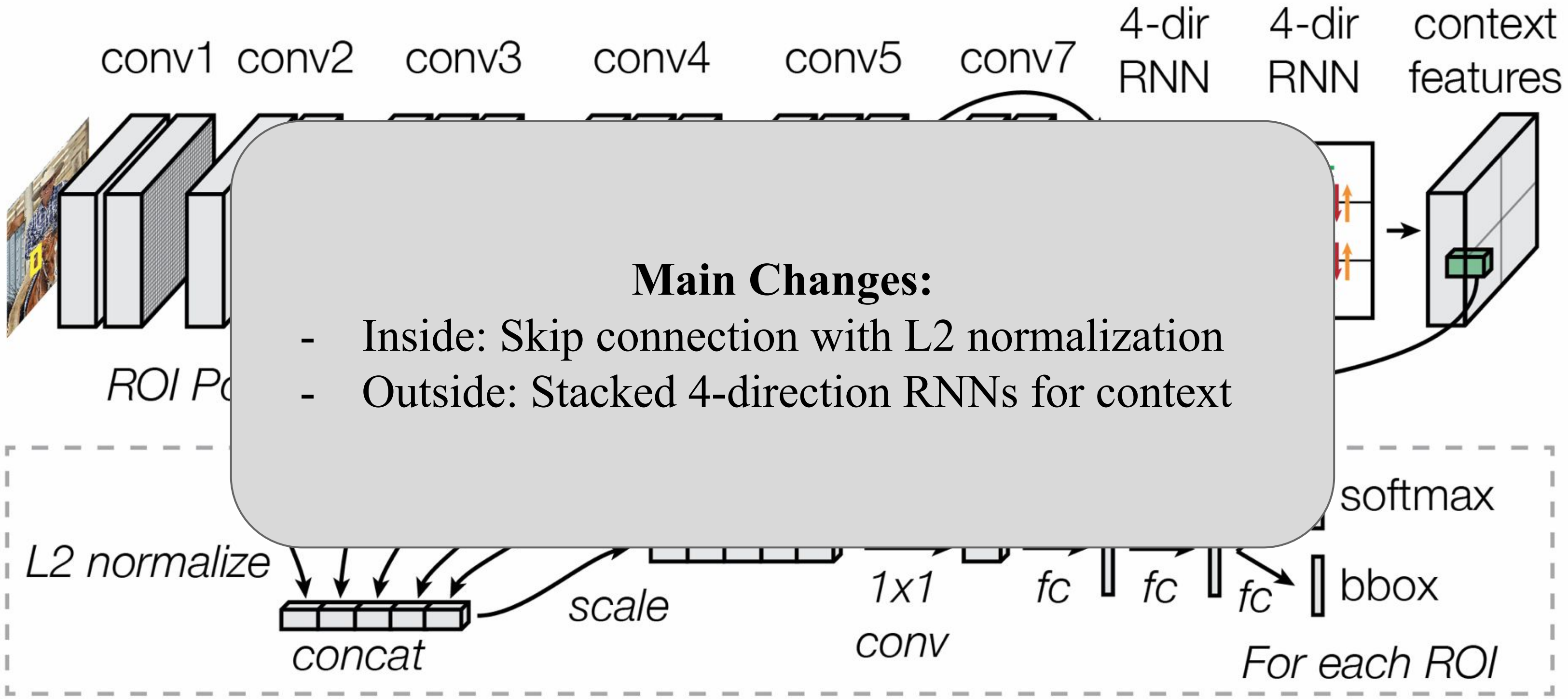
Uses RNNs to capture context info from outside bounding box.



Stack 2 RNNs together



ION: Inside-Outside Net



Main Changes:

- Inside: Skip connection with L2 normalization
- Outside: Stacked 4-direction RNNs for context

Results (VOC 2007)

METHOD	MAP
FAST R-CNN [GIRSHICK 2014]	70.0
FASTER R-CNN [GIRSHICK 2015]	73.2
CONV3+CONV4+CONV5	**75.6**
+ RNN + SEGMENTATION LOSS	**76.5**
+ SECOND BBOX REGRESSION + WEIGHTED VOTING	**78.5**
— DROPOUT	**79.2**

https://www.robots.ox.ac.uk/~vgg/rg/slides/ion-coco.pdf

Trained on Pascal VOC 2007 + 2012 dataset

Summary and Comparison

Part IV

Speed / Accuracy Trade-off

Unified tensorflow architecture

Compare speed, accuracy and memory usage

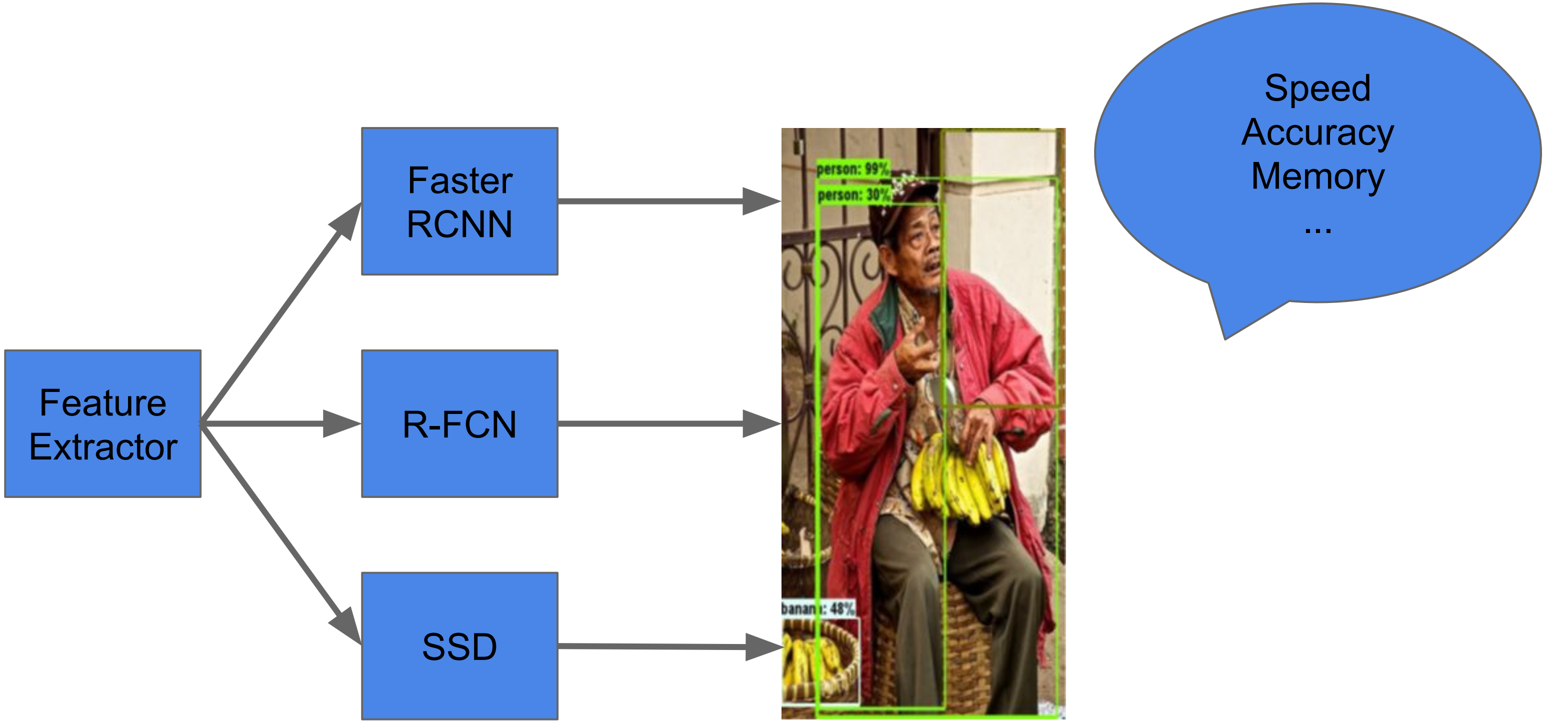
Speed/Accuracy trade-offs for modern convolutional object detectors

Jonathan Huang, Kevin Murphy et al.

Nov 2016

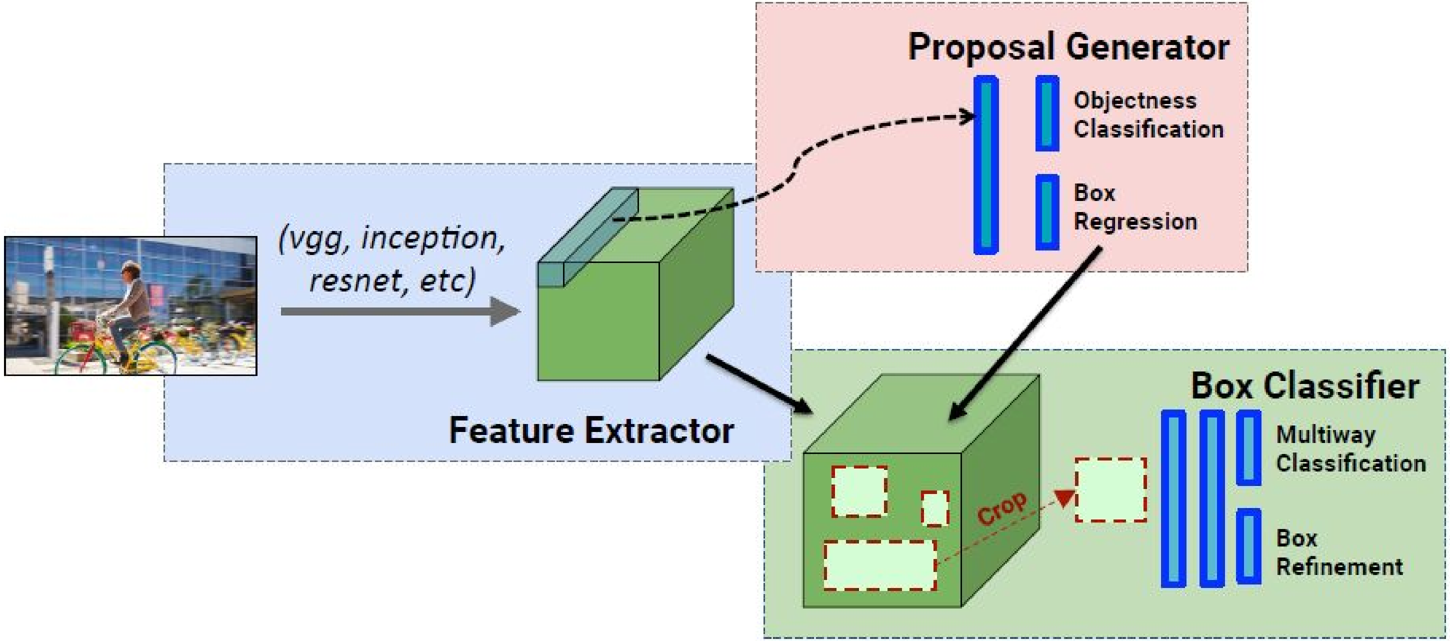
https://people.eecs.berkeley.edu/~rbg/papers/r-cnn-cvpr.pdf

Same Architectures



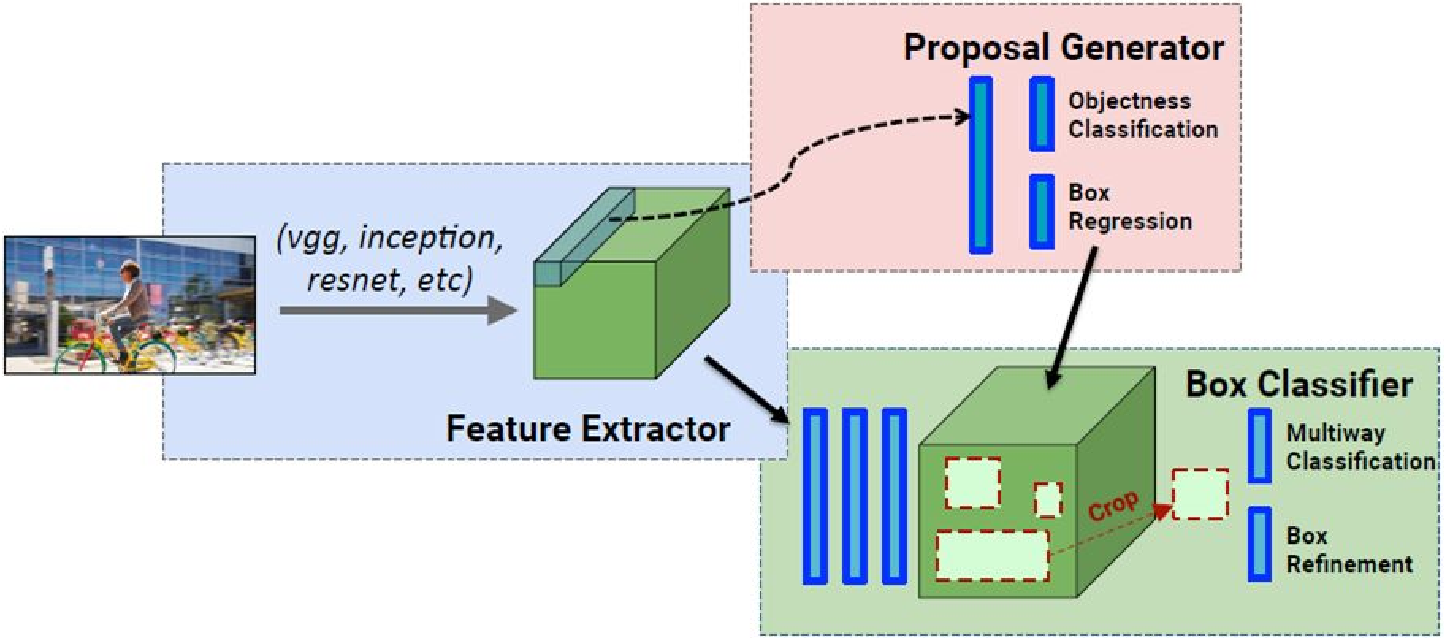
https://pan.baidu.com/s/1pKIKIIB

Recap: Faster RCNN



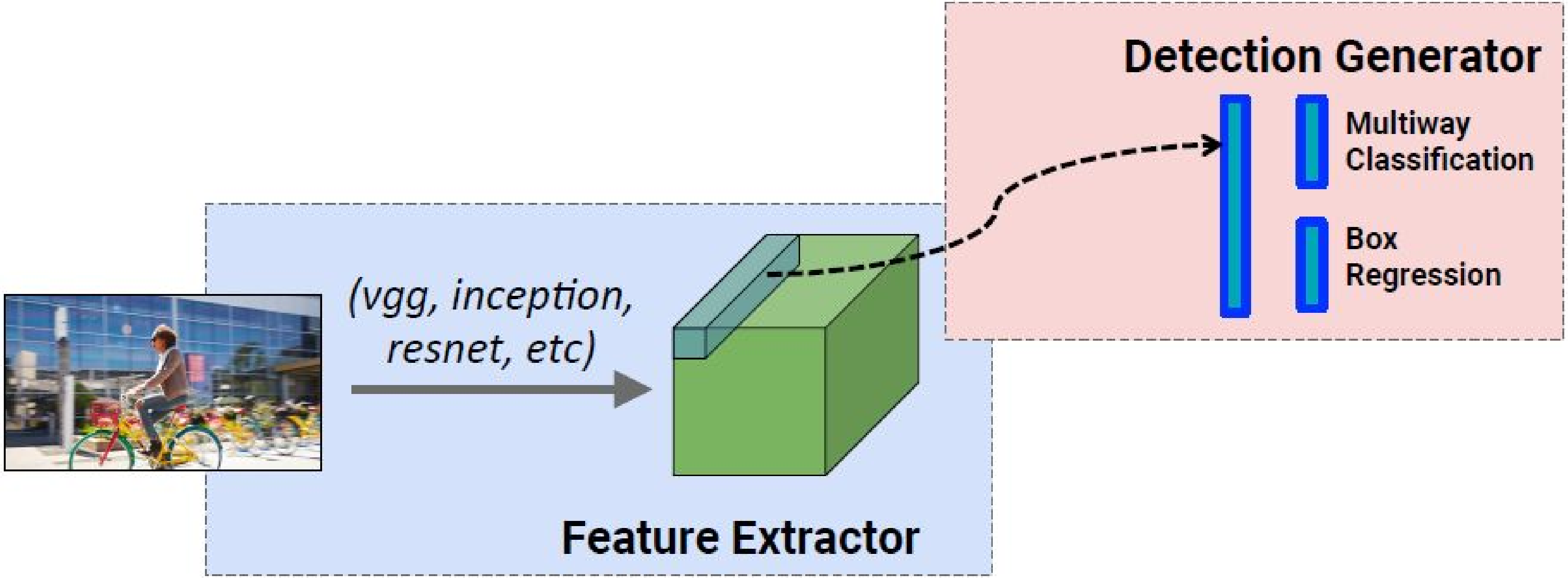
https://pan.baidu.com/s/1pKIKIIB

Recap: R-FCN



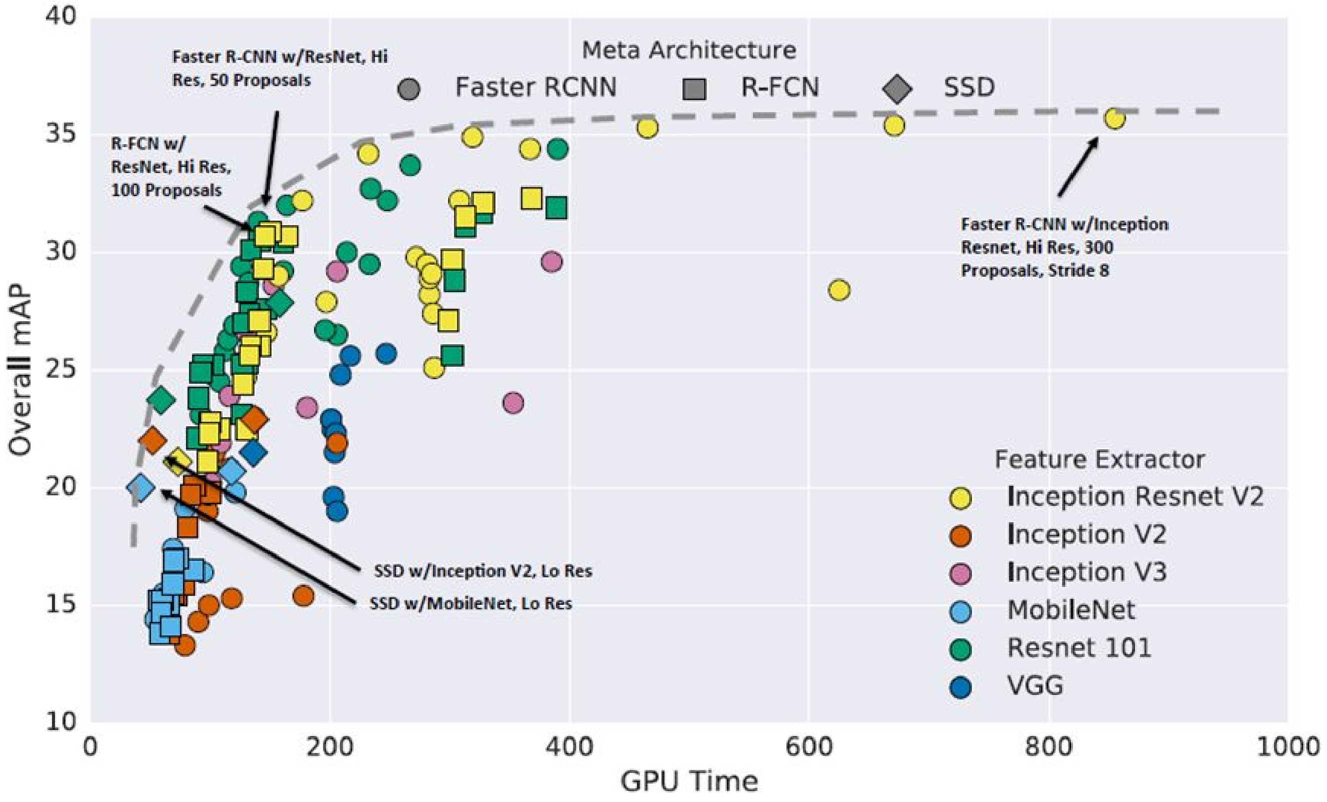
https://pan.baidu.com/s/1pKIKIIB

Recap: SSD

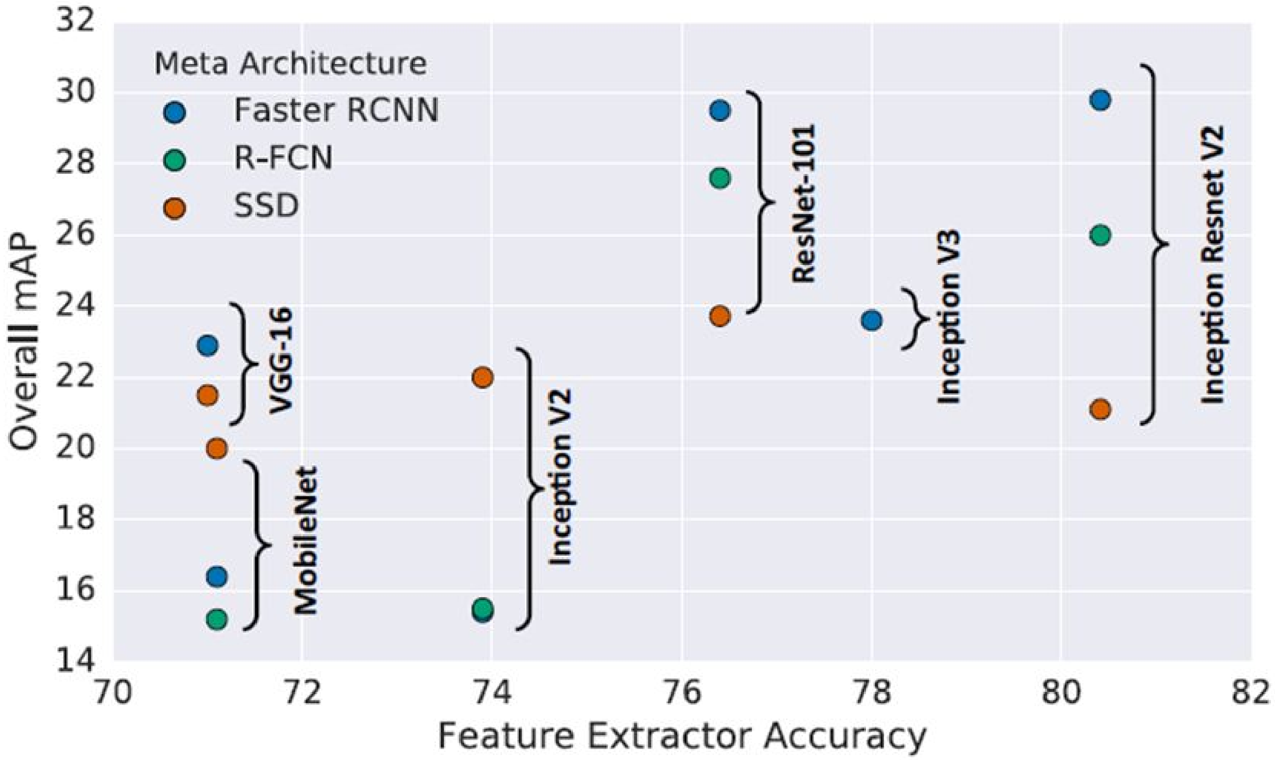


https://pan.baidu.com/s/1pKIKIIB

Accuracy VS Speed



1. Different Feature Extractor



2. Detect Object Size

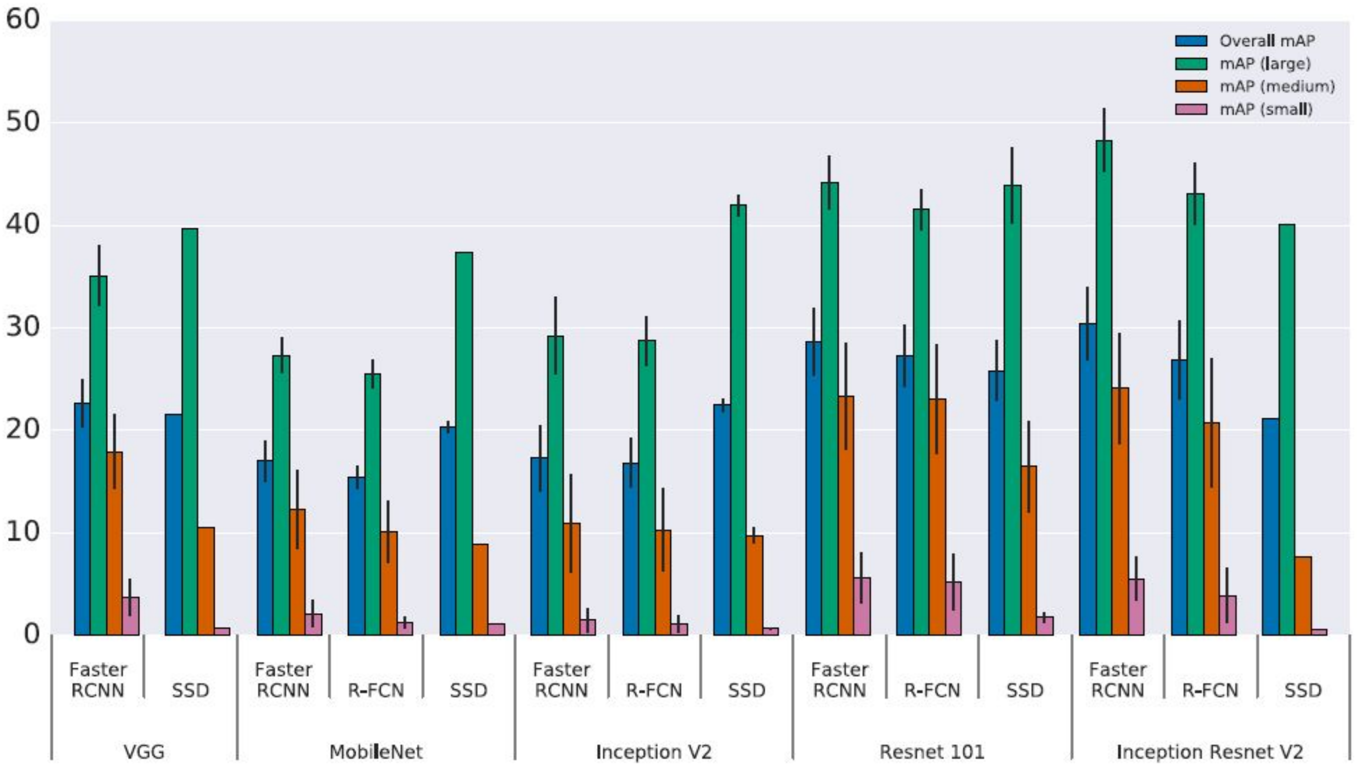
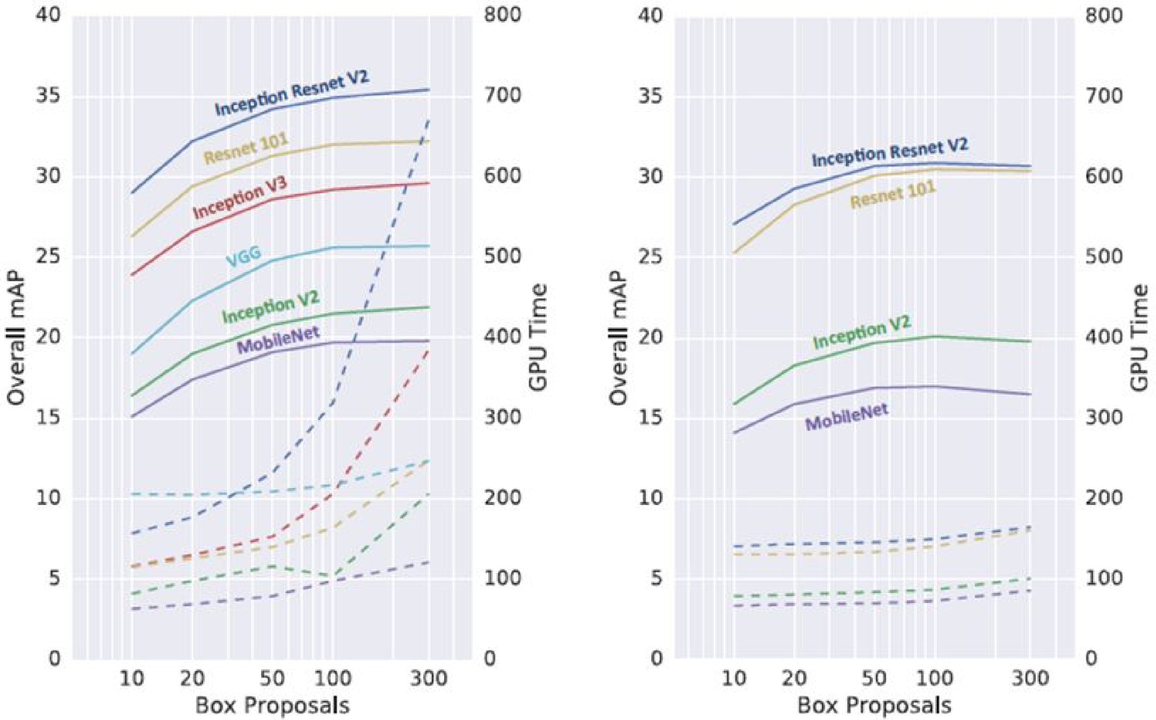


Figure 4: Accuracy stratified by object size, meta-architecture and feature extractor. We fix the image resolution to 300.

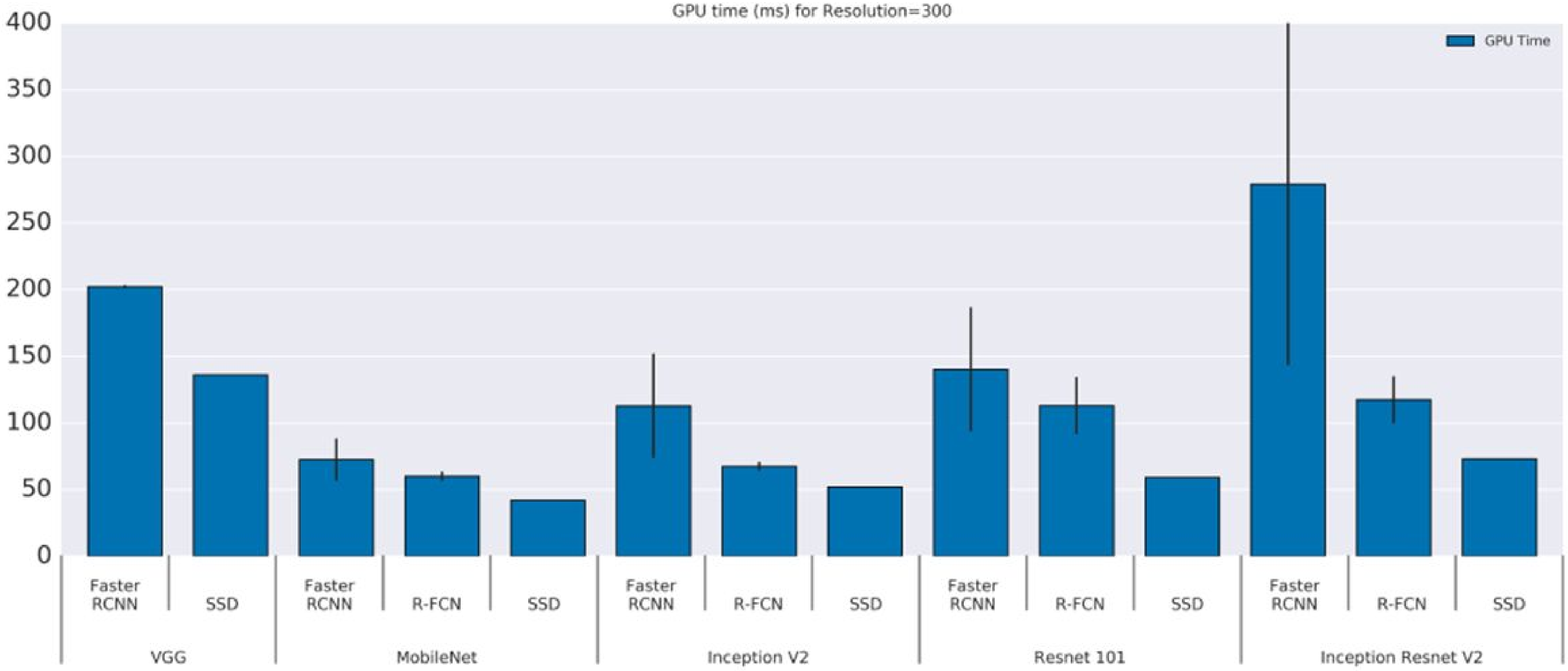
3. Image Resolution

4. Region Proposal Number

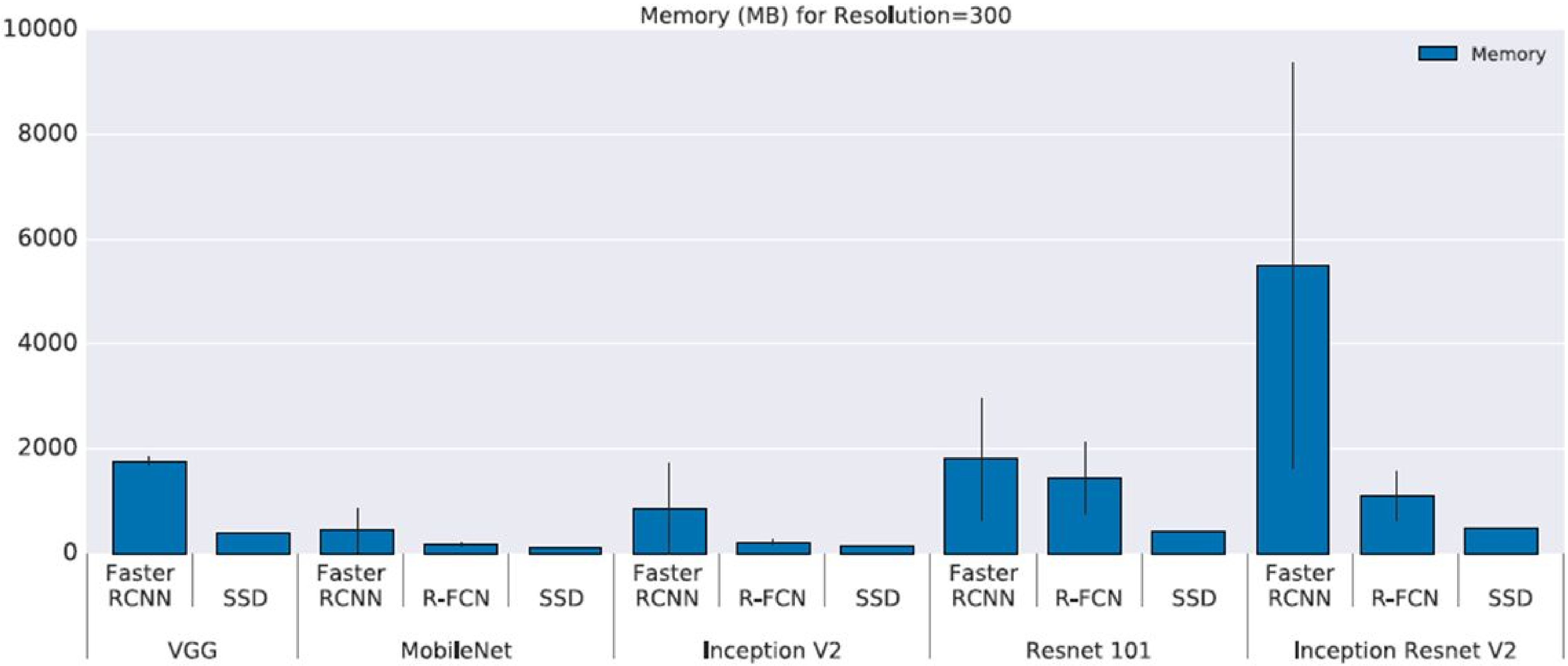


(a) FRCNN (b) RFCN

5. GPU Time


GPU time (ms) for Resolution=300
GPU Time
400
350
300
250
200
150
100
50
0
Faster RCNN
SSD
Faster RCNN
R-FCN
SSD
Faster RCNN
R-FCN
SSD
Faster RCNN
R-FCN
SSD
Faster RCNN
R-FCN
SSD
VGG
MobileNet
Inception V2
Resnet 101
Inception Resnet V2

6. Memory



Memory (MB) for Resolution=300
10000
8000
6000
4000
2000
0
Memory
Faster RCNN
SSD
Faster RCNN
R-FCN
SSD
Faster RCNN
R-FCN
SSD
Faster RCNN
R-FCN
SSD
Faster RCNN
R-FCN
SSD
VGG
MobileNet
Inception V2
Resnet 101
Inception Resnet V2

Summary

- Speed First: SSD

- Balance Speed and Accuracy: R-FCN

- Accuracy First: Faster RCNN (Reduce proposals, it can speed up a lot with some accuracy loss)

Thanks!

Questions?

References

Girshick, Ross and Donahue, Jeff and Darrell, Trevor and Malik, Jitendra, Rich feature hierarchies for accurate object detection and semantic segmentation, CVPR 2014

He, Kaiming and Zhang, Xiangyu and Ren, Shaoqing and Sun, Jian, Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition, ECCV 2014

Girshick, Ross, Fast R-CNN, ICCV 2015

Ren, Shaoqing and He, Kaiming and Girshick, Ross and Sun, Jian, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, CVPR 2015

Jifeng Dai, Yi Li, Kaiming He, Jian Sun R-FCN: Object Detection via Region-based Fully Convolutional Networks, NIPS 2016

Erhan, Dumitru and Szegedy, Christian and Toshev, Alexander and Anguelov, Dragomir, Scalable Object Detection using Deep Neural Networks, CVPR 2014

Bell, Sean and Lawrence Zitnick, C and Bala, Kavita and Girshick, Ross, Inside-Outside Net: Detecting Objects in Context with Skip Pooling and Recurrent Neural Networks, CVPR 2016

Redmon, Joseph and Divvala, Santosh and Girshick, Ross and Farhadi, Ali, You Only Look Once: Unified, Real-Time Object Detection, CVPR 2016

Liu, Wei and Anguelov, Dragomir and Erhan, Dumitru and Szegedy, Christian and Reed, Scott and Fu, Cheng-Yang and Berg, Alexander C, SSD: Single Shot MultiBox Detector, ECCV 2016

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, Serge Belongie, Feature Pyramid Networks for Object Detection, arXiv 2016

Huang, Jonathan and Rathod, Vivek and Sun, Chen and Zhu, Menglong and Korattikara, Anoop and Fathi, Alireza and Fischer, Ian and Wojna, Zbigniew and Song, Yang and Guadarrama, Sergio and others,Speed/accuracy trade-offs for modern convolutional object detectors, arXiv 2016

Bonus Material

Part V

Yolo v2 Faster, Better Stronger

	YOLO								YOLOv2
batch norm?		✓	✓	✓	✓	✓	✓	✓	✓
hi-res classifier?			✓	✓	✓	✓	✓	✓	✓
convolutional?				✓	✓	✓	✓	✓	✓
anchor boxes?				✓	✓				
new network?					✓	✓	✓	✓	✓
dimension priors?						✓	✓	✓	✓
location prediction?						✓	✓	✓	✓
passthrough?							✓	✓	✓
multi-scale?								✓	✓
hi-res detector?									✓
VOC2007 mAP	63.4	65.8	69.5	69.2	69.6	74.4	75.4	76.8	**78.6**

Performance on MS COCO

Method	Data	Avg. Precision						Avg. Recall					
		IoU			Area			#Dets			Area		
		0.5:0.95	0.5	0.75	S	M	L	1	10	100	S	M	L
Fast R-CNN	train	19.7	35.9	-	-	-	-	-	-	-	-	-	-
Fast R-CNN	train	20.5	39.9	19.4	4.1	20.0	35.8	21.3	29.5	30.1	7.3	32.1	52.0
Faster R-CNN	trainval	21.9	42.7	-	-	-	-	-	-	-	-	-	-
ION	train	23.6	43.2	23.6	6.4	24.1	38.3	23.2	32.7	33.5	10.1	37.7	53.6
Faster R-CNN	trainval	24.2	45.3	23.5	7.7	26.4	37.1	23.8	34.0	34.6	12.0	38.5	54.4
SSD300	trainval35k	23.2	41.2	23.4	5.3	23.2	39.6	22.5	33.2	35.3	9.6	37.6	56.5
SSD512	trainval35k	**26.8**	**46.5**	**27.8**	**9.0**	**28.9**	**41.9**	**24.8**	**37.5**	**39.8**	**14.0**	**43.5**	**59.0**
YOLOv2	trainval35k	21.6	44.0	19.2	5.0	22.4	35.5	20.7	31.6	33.3	9.8	36.5	54.4

Fast RCNN vs Faster RCNN

method	# proposals	data	mAP (%)
SS	2000	07	66.9†
SS	2000	07+12	70.0
RPN+VGG, unshared	300	07	68.5
RPN+VGG, shared	300	07	69.9
RPN+VGG, shared	300	07+12	**73.2**
RPN+VGG, shared	300	COCO+07+12	**78.8**

SPP-Net

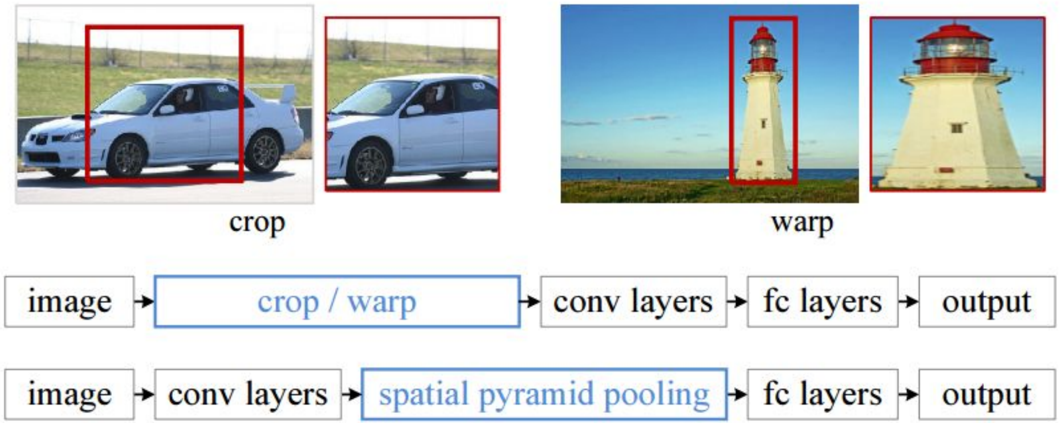
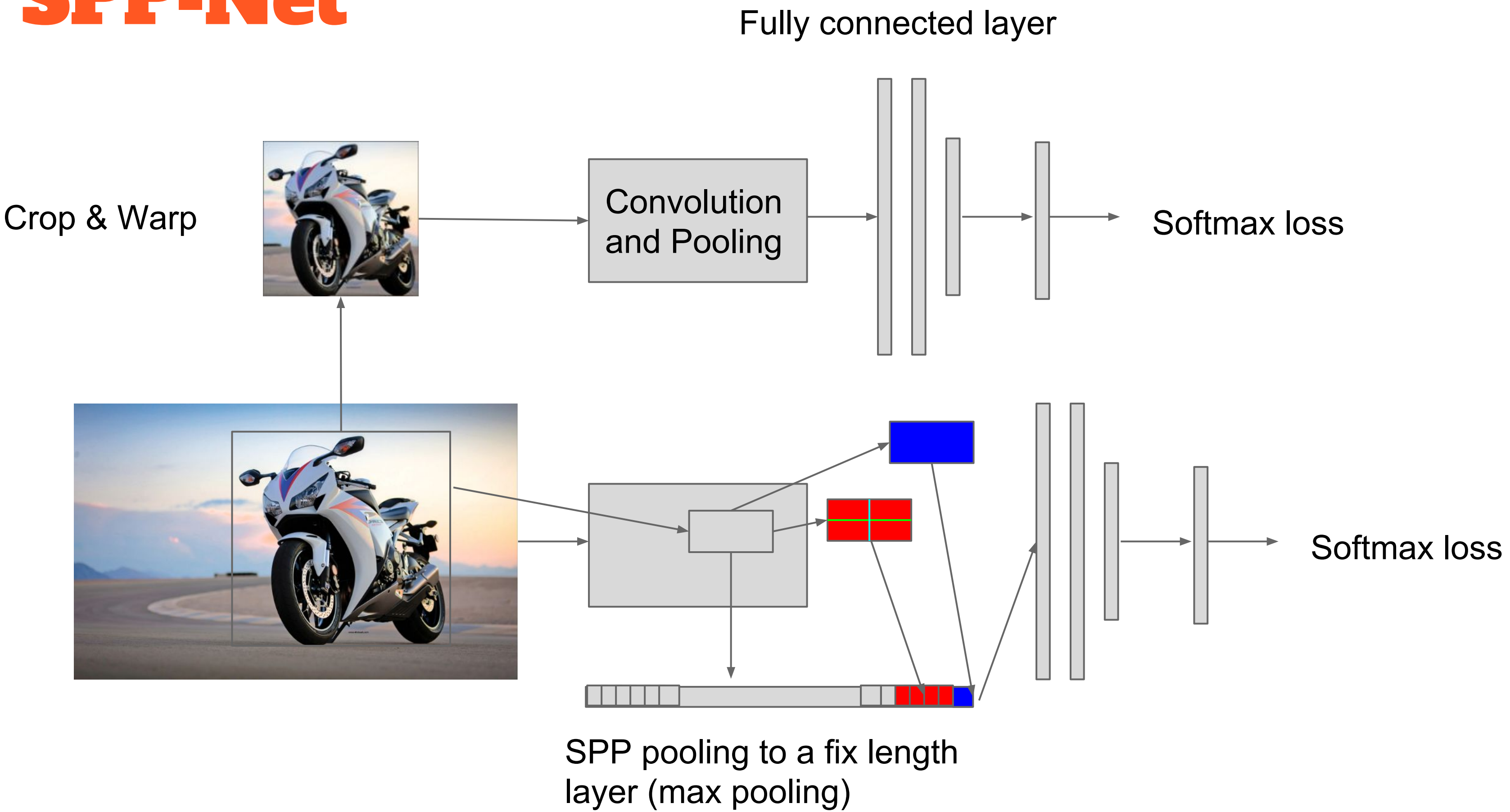


Figure 1: Top: cropping or warping to fit a fixed size. Middle: a conventional CNN. Bottom: our spatial pyramid pooling network structure.

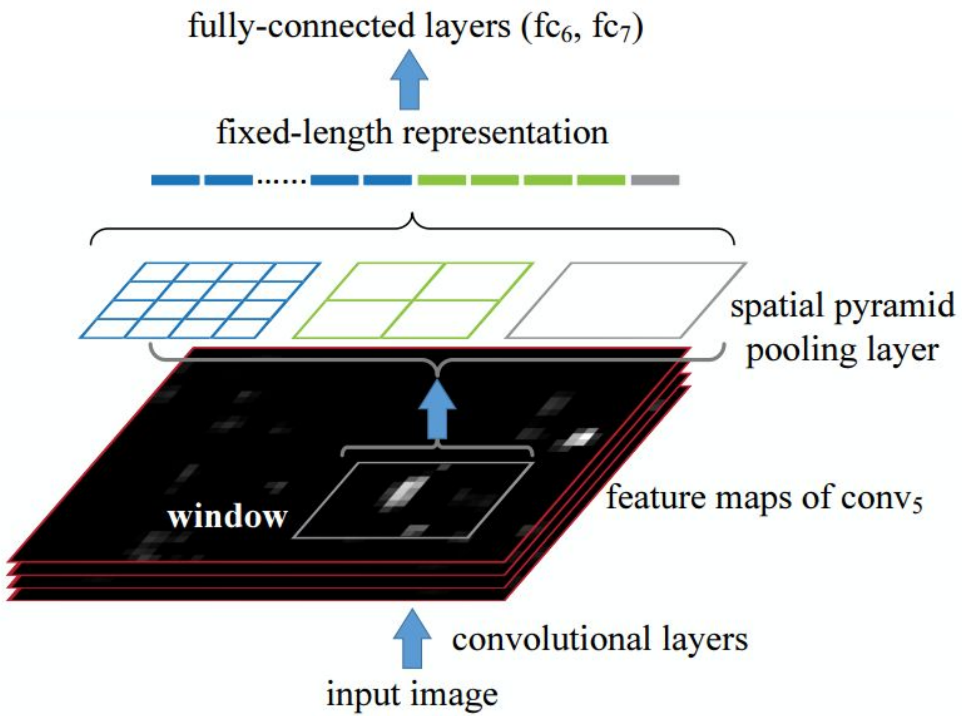
https://arxiv.org/pdf/1406.4729v4.pdf

SPP-Net



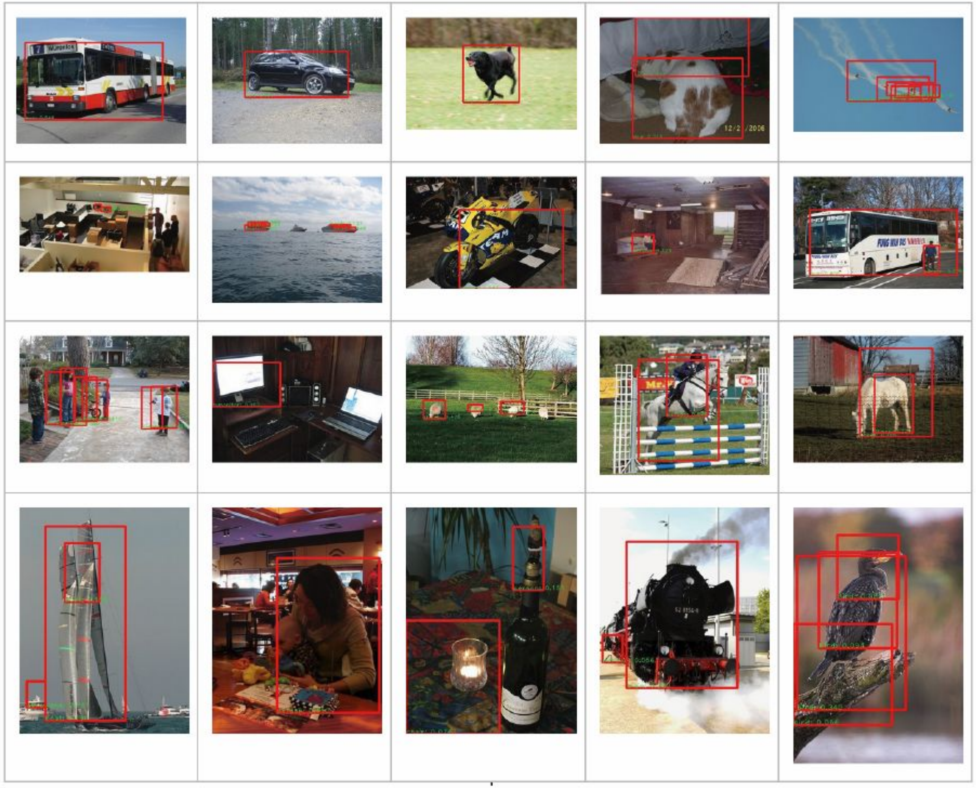
Fully connected layer
Crop & Warp
Convolution and Pooling
Softmax loss
Softmax loss
SPP pooling to a fix length layer (max pooling)

SPP-Net



https://arxiv.org/pdf/1406.4729v4.pdf

Multi-Box



Multi-Box

Goal: Achieve a class-agnostic scalable object detection by predicting a set of bounding boxes.

Train a neural network to directly predict:

- The upper-left and lower-right coordinates of each bounding box
- The confidence score for the box containing an object

Train Objective

Produce fixed number of bounding boxes with confidence, such as K=100 or 200.

$$F(x, l, c) = \alpha F_{\text{match}}(x, l) + F_{\text{conf}}(x, c)$$

$$\begin{aligned} x^* &= \arg\min_x F(x, l, c) \\ \text{subject to} \quad & x_{ij} \in \{0, 1\}, \sum_i x_{ij} = 1, \end{aligned}$$

Number of Windows

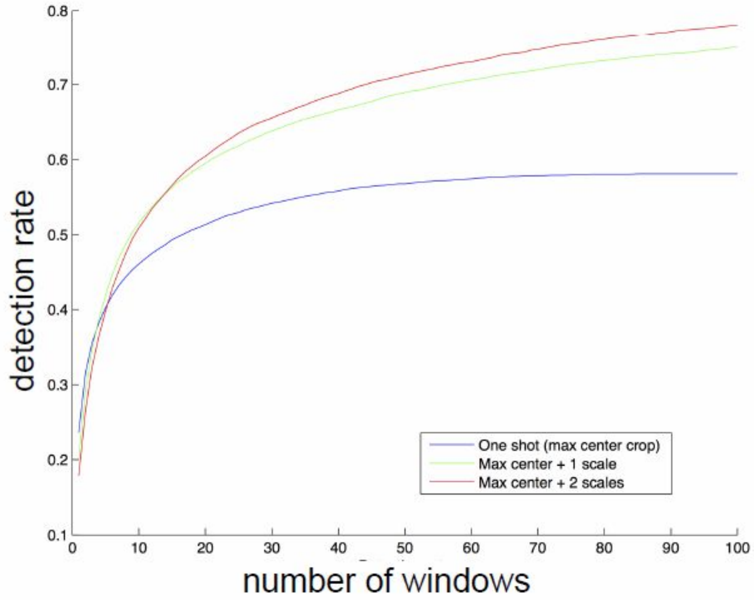


Figure 1. Detection rate of class "object" vs number of bounding boxes per image. The model, used for these results, was trained on VOC 2012.