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

Shaoqing Ren, Kaiming He,
Ross Girshick, Jian Sun

IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016

Outline

- Goal
- Methodology
  - Foundation (R-CNN and Fast R-CNN)
  - Main Approach
- Experimental Results
  - Quantitative
  - Qualitative

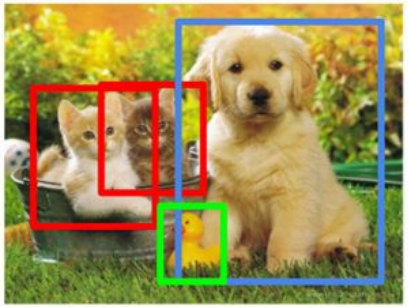
Goal

Object detection → **Find** and **classify** objects in an image with applications in:

- Autonomous Driving
- Surveillance systems
- Industry

Success measurements:

- Accuracy
- Speed

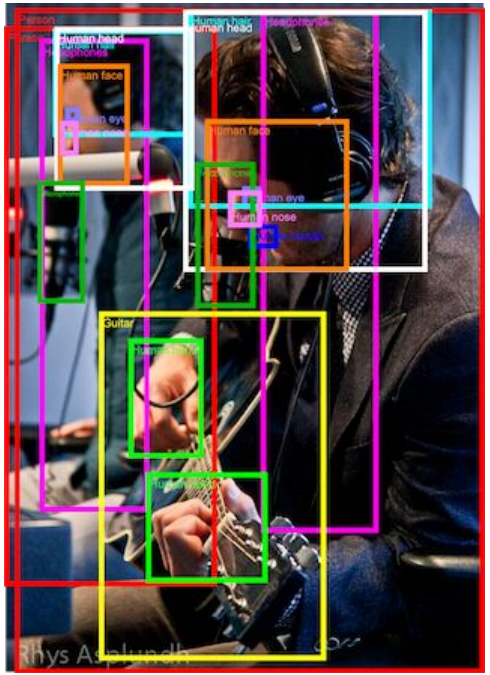


CAT, DOG, DUCK

Goal

Some challenges:

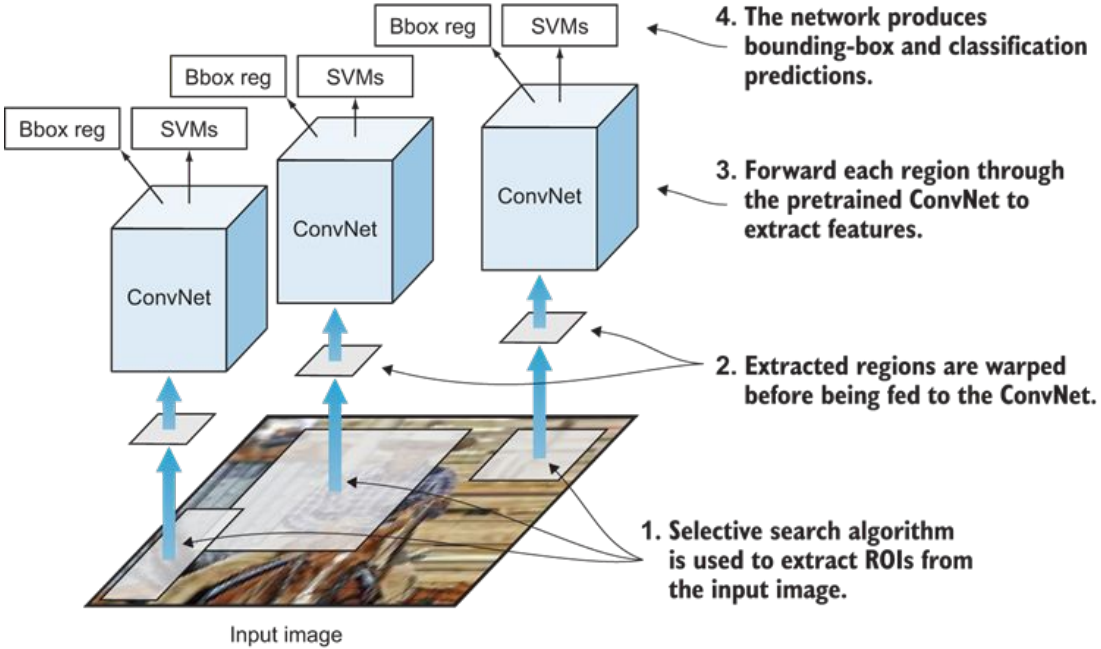
- Occlusion/Self occlusion
- Different viewpoints
- High variety of appearance within a class type
- Multiple scales
- Multiple aspect ratio



Open Images 2019 - Object Detection

Methodology - Foundation

R-CNN

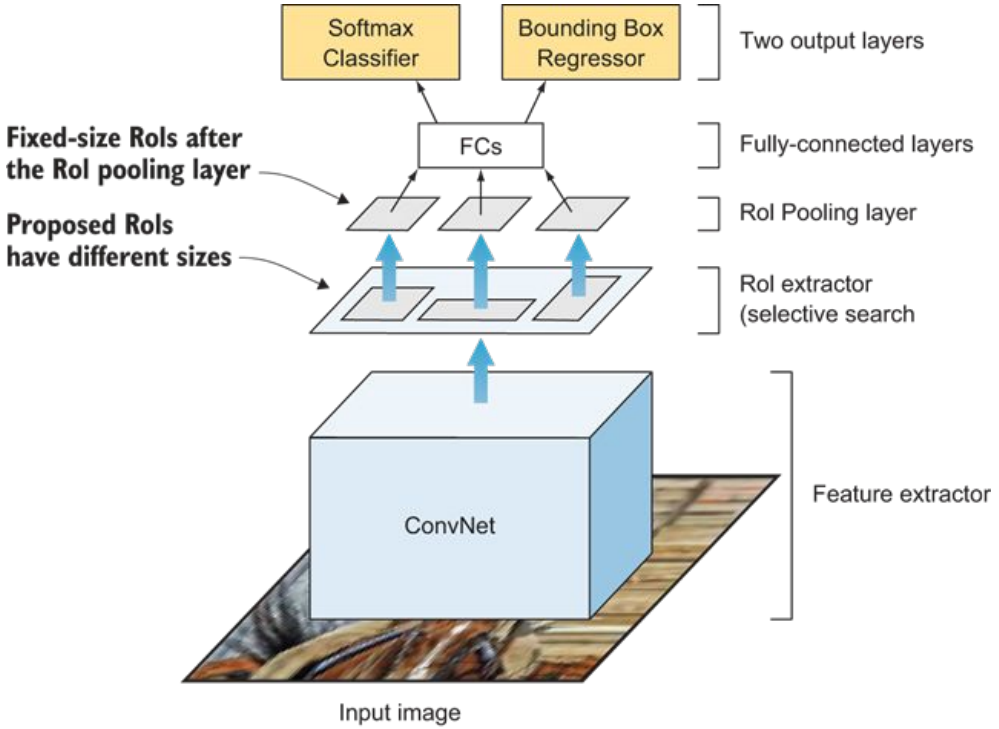


"Deep Learning for Vision Systems", Mohamed Elgendy

"Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", Ross Girshick et al., CVPR 2014

Methodology - Foundation

Fast R-CNN

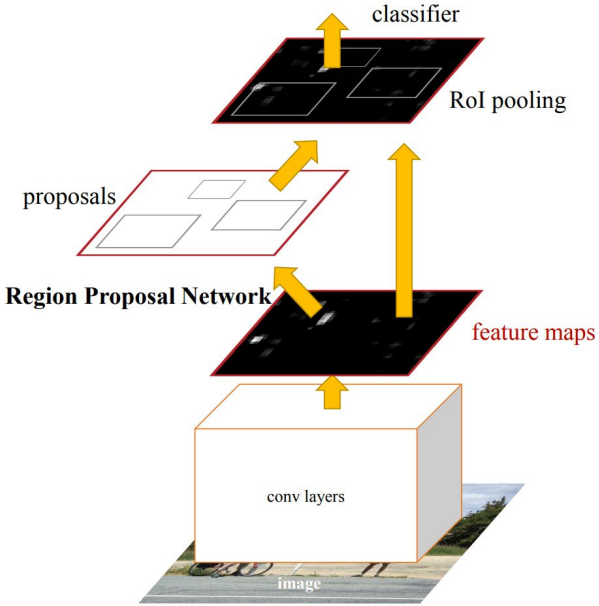


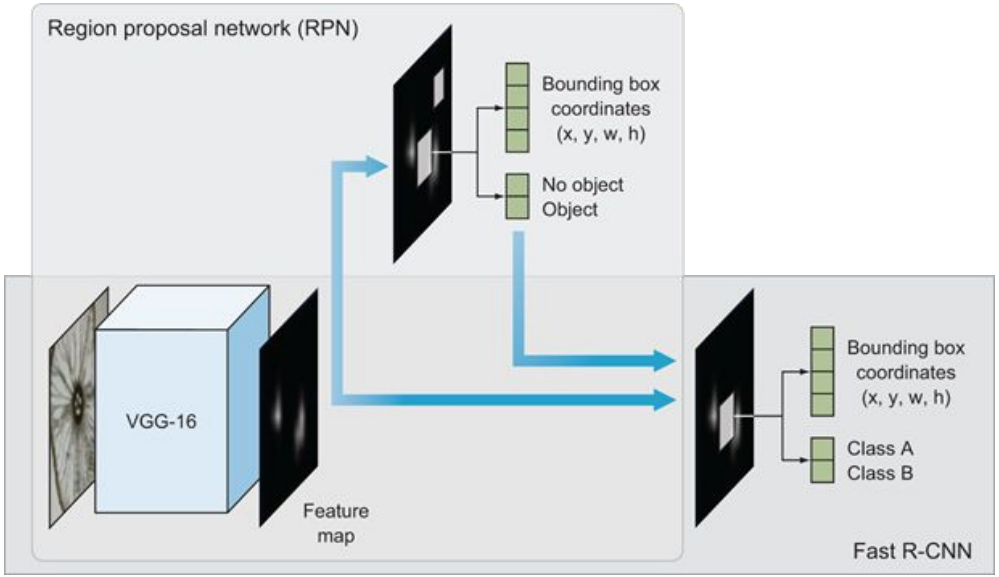
"Deep Learning for Vision Systems", Mohamed Elgendy

"Fast R-CNN", Ross Girshick, ICCV 2015

Methodology - Main Approach

Faster R-CNN





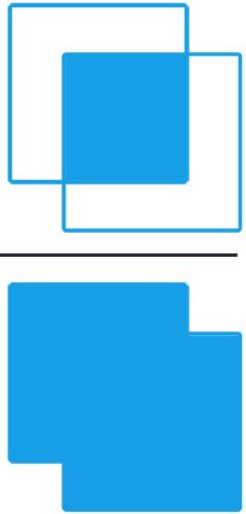
"Deep Learning for Vision Systems", Mohamed Elgendy

Methodology - Main Approach

Region Proposal Network (RPN)

"Anchors" with success metric as IoU (**I**ntersection **o**ver **U**nion)

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$



$$IoU = \begin{cases} > 0.7 & \text{positive} \\ < 0.3 & \text{negative} \end{cases}$$

Methodology - Main Approach

Region Proposal Network (RPN)

Feature map: ***H×W×D*** → ***H×W*** anchors of size **1×*D***

Multi-scale predictions → **9** Anchors: **3** Ratios × **3** Resolutions

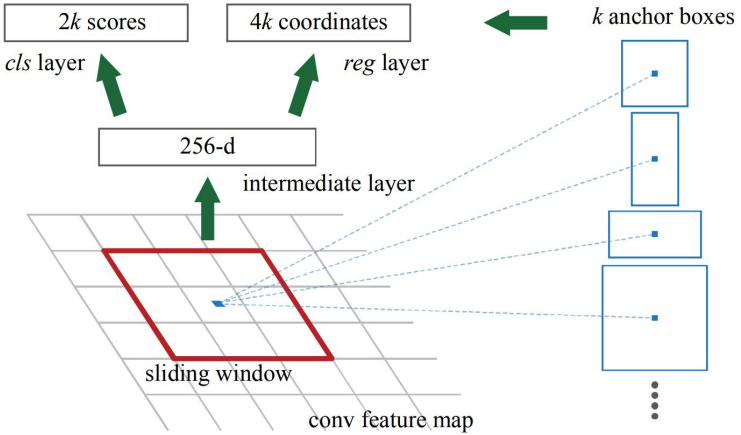
1:1, 1:2, 2:1 128, 256, 512

40×60×9 =**21.6k** → Ignoring cross boundary: ~**6k**

→ Non-maximum suppression: **~2k**

→ randomly sample **256** samples for loss computation

Methodology - Main Approach



Region Proposal Network (RPN)

Predictions via for each anchor:

- Objectness score (object/non-object probability)
- Bounding box regression, [*x*,*y*,*w*,*h*]

Output size: (2+4) × 9

Loss function:

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*)$$

Methodology - Main Approach

Alternating Training

1. Train **RPN** on **ImageNet pretrained model**

2. Train **Fast R-CNN** using *proposals generated by step 1* on **ImageNet pretrained model**

----no sharing up to this point----

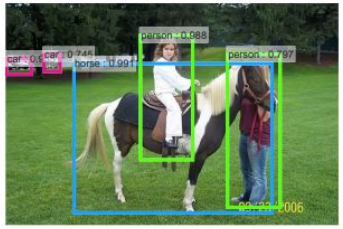
3. With the frozen CNN from step 2, fine-tune the layers of **RPN**

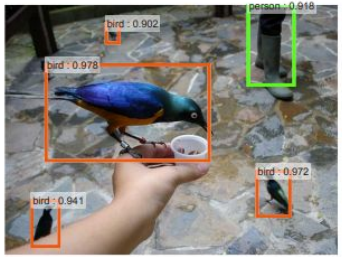
4. With the frozen network from step 3, fine-tune the layers of **Fast R-CNN** using *proposals generated by RPN*

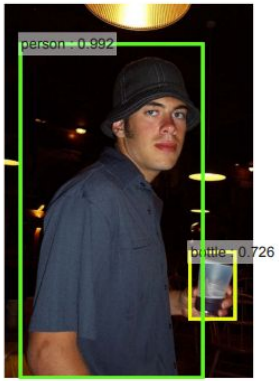
Experimental Results

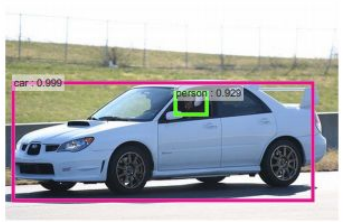
	# of proposals	mAP@0.5 (%) **Pascal VOC 2007**	mAP@0.5 (%) **Microsoft COCO**	rate (fps)
Fast R-CNN (VGG)	2000	70.0	39.3	0.5
Faster R-CNN (VGG)	300	**73.2**	**42.1**	**5**

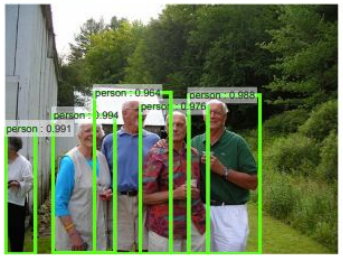
Experimental Results

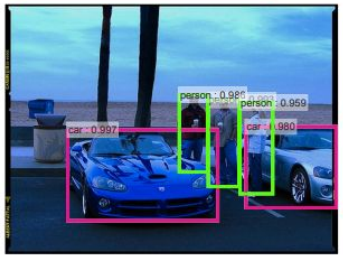


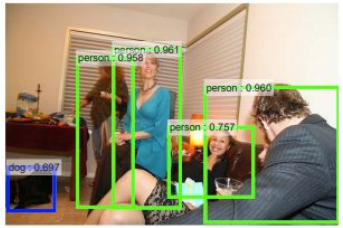


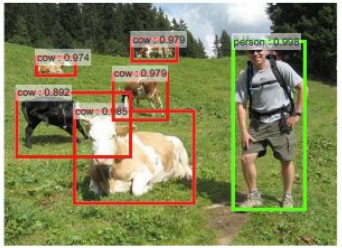






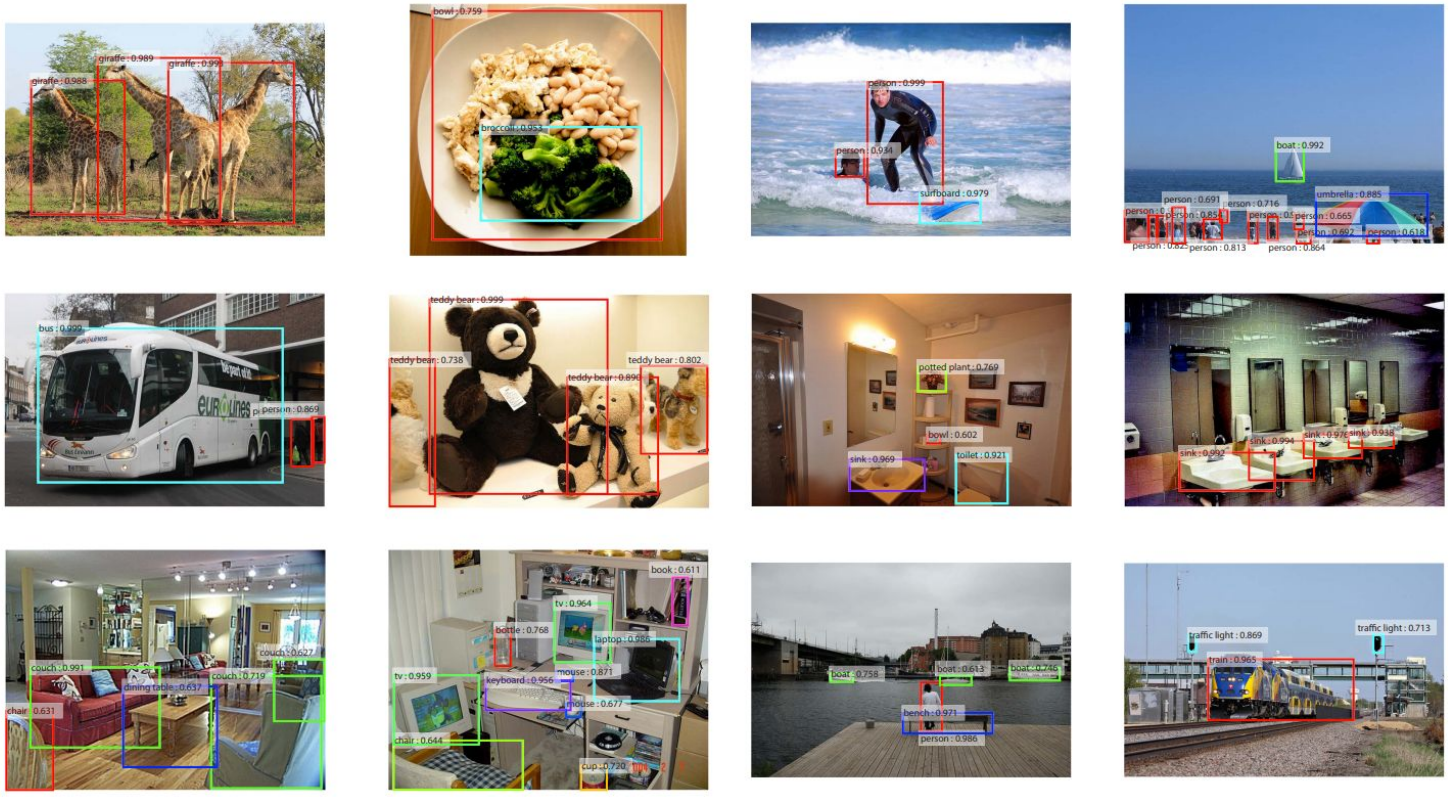






Pascal VOC 2007

Experimental Results



MS-COCO test