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

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

Presented by Riaan Zoetmulder and Minh Ngo

University of Amsterdam

April 25, 2016

Authors

- Shaoqing Ren, PhDs, USTC & Microsoft Research Asia
- Kaiming He, Lead Researcher, Microsoft Research Asia
  - Deep Residual Learning for Image Recognition (2015) `http://doi.org/10.3389/fpsyg.2013.00124` (state-of-the art!)
- **Ross Girshick**, Research Scientist, Facebook AI Research
  - The author of R-CNN papers
- Jan Sun, Principal Research Manager, Microsoft Research
  - CaptionBot `http://captionbot.ai/`
  - Rich Image Captioning in the wild (2016) `http://research.microsoft.com/pubs/264408/ImageCaptionInWild.pdf`
  - Deep Residual Learning for Image Recognition (2015)

R-CNN Timeline

- Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 580587. http://doi.org/10.1109/CVPR.2014.81
- Girshick, R. (2015). Fast R-CNN. http://doi.org/10.1109/ICCV.2015.169
- Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. ArXiv 2015, 110. http://doi.org/10.1016/j.nima.2015.05.028

- ???

Object Detection Task Pipeline

- Bounding box proposal
- Feature Extraction for bounding box proposals
- Classification based on features

R-CNN (Region based CNN)

- Region proposal (Selective Search)
- Resize image regions to 227x227, mean subtracted
- Fixed length feature vector extracted by CNN (pretrained AlexNet) $\forall$ region
- Class specific SVMs
- Non-Maximum Supression

R-CNN (Region based CNN)

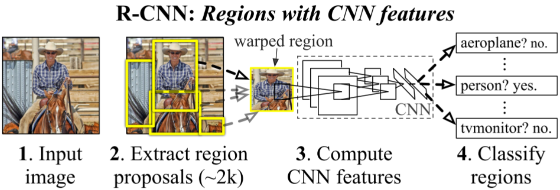

Figure: R-CNN architecture. Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation.

R-CNN (Region based CNN)

Contribution of the paper

- Combining region proposals with CNN
- Training a large CNN on a small amount of labeled data
  - Unsupervised pre-training on the large dataset annotated on image level (without bounding boxes)
  - Supervised fine-tuning with smaller domain-specific dataset

R-CNN (Region based CNN)

Advantage:

- CNN parameters are shared accross $\forall$ categories
- Low dimensional features

Disadvantage:

- Features are required to be stored for SVMs
- Features for different bounding box proposals are extracted separately

Fast R-CNN

- Faster training / testing (x10 - x100 faster than R-CNN)
- More accurate
- Single stage training algorithm (no separate feature extraction!) that joinly learns
  - to classify object proposals
  - to refine spatial locations
- Sharing ConvNet features computation for $\forall$ object proposals

Fast R-CNN

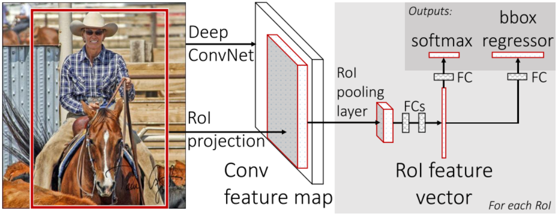

Figure: Fast R-CNN Architecture. Girshick, R. (2015). Fast R-CNN.

Fast R-CNN

- Input: Entire image
- Processes a whole image with several convolutions & max pooling layers to produce a convolution feature map (state-of-the-art models like VGG can be used)
- Object proposals (from Selective Search) and a feature map are put into the RoI pooling layer
- Fully Connected Layers
- Softmax $(K + 1)$ classes (number of objects + background)
- Four real valued numbers (refined bounding box proposals) $\forall$ object classes

Region-of-Interest Pooling

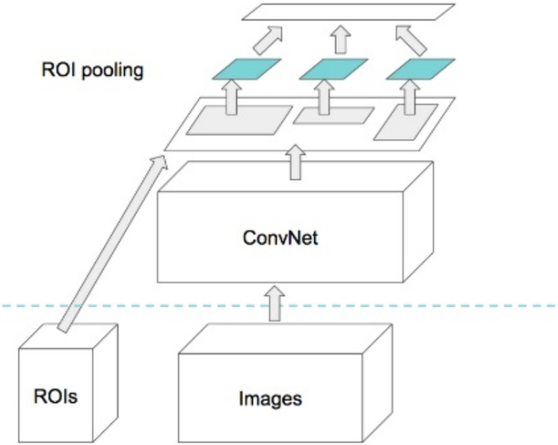

Figure: RoI. (c) Yinyin Liu 2016

Region-of-Interest Pooling

- Hyper parameters: N,D
- Arbitrary size of input
- Divide input into a grid of NxD
- Do Max Pooling for each cell

Faster R-CNN

Contributions of the paper:

- Incorporates the bounding box proposal part into the neural network architecture (Region proposal network)
- Even faster (5 fps on GPU).
- A new scheme for addressing objects of multiple scales and sizes using pyramids of reference boxes

Faster R-CNN

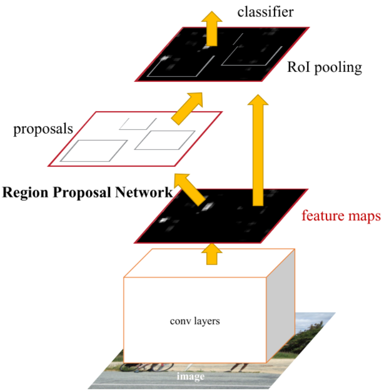

Figure: Faster R-CNN architecture. Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.

Faster R-CNN

Two modules:

- Deep CNN that proposes regions (Region proposal network)
- Fast R-CNN detector that uses the proposed regions ("Attention" mechanism).

Region Proposal Network

- Works like "sliding window"
- Input: Arbitrary size image
- Output: $\forall$ rectangular proposal 4 bounding box coordinates (regression layer) and 2 objectness scores (classification later)
- Shares the Convolution Layer with the object detection block.

Addressing multiple scales and sizes

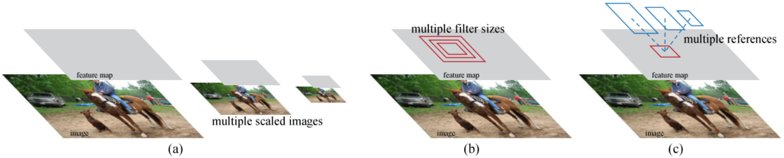

Figure: Different schemes for addressing multiple scales and sizes. Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.

Anchors

K region proposals (Anchors) $\forall$ sliding window locations.

- Centered at the sliding window
- 3 scales, 3 aspect ratios
- Translation-invariant

Anchors

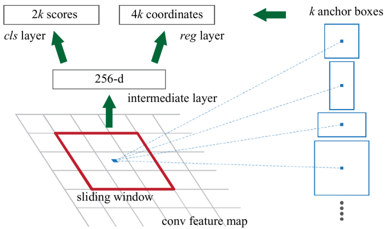

Figure: Region Proposal Network. Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.

Demo time

- https://www.youtube.com/watch?v=u5W05Ej1HBg
- https://www.youtube.com/watch?v=0TWvtjLPwNc
- Fast R-CNN
  https://www.youtube.com/watch?v=6HHkf1AQZ_c

Loss Function

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

where,
p_i = predicted probability anchor is object
p_i^* = 1 iff anchor is positive, 0 otherwise
t_i = vector representation of coordinates
t_i^* = ground truth box associated with positive anchor.
L_{cls} = log loss
$L_{reg} = R(t_i - t_i^*)$

RPN-Training

- Mini batch of 1 image
- get sample anchors
- compute the loss function
- if positive examples smaller than 128, pad with negative examples.

Feature sharing for RPN and Fast R-CNN

In this paper they used alternate training for to train RPN and the fast R-CNN.

- First, train RPN.
- Use proposals to train Fast R-CNN
- Use Fast R-CNN to initialize RPN again.
- Keep shared Convolutional layers fixed. Train layers of RCNN.

Feature sharing for RPN and Fast R-CNN

(Use drawings on blackboard to clarify.)

Feature sharing for RPN and Fast R-CNN

so now we have a lot of overlapping regions. How do we reduce redundancy?

Experiments

- Pascal VOC
  - ablation experiments
  - Performance of VGG-16
  - Sensitivity to Hyper parameters
  - Analysis of recall to IOU
  - One stage detection vs two stage detection
- experiments on MS COCO

Ablation experiments

Table 2: Detection results on **PASCAL VOC 2007 test set** (trained on VOC 2007 trainval). The detectors are Fast R-CNN with ZF, but using various proposal methods for training and testing.

train-time region proposals		**test-time region proposals**		
method	# boxes	method	# proposals	mAP (%)
SS	2000	SS	2000	58.7
EB	2000	EB	2000	58.6
RPN+ZF, shared	2000	RPN+ZF, shared	300	**59.9**
ablation experiments follow below				
RPN+ZF, unshared	2000	RPN+ZF, unshared	300	58.7
SS	2000	RPN+ZF	100	55.1
SS	2000	RPN+ZF	300	56.8
SS	2000	RPN+ZF	1000	56.3
SS	2000	RPN+ZF (no NMS)	6000	55.2
SS	2000	RPN+ZF (no *cls*)	100	44.6
SS	2000	RPN+ZF (no *cls*)	300	51.4
SS	2000	RPN+ZF (no *cls*)	1000	55.8
SS	2000	RPN+ZF (no *reg*)	300	52.1
SS	2000	RPN+ZF (no *reg*)	1000	51.3
SS	2000	RPN+VGG	300	59.2

performance of VGG-16

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**

Sensitivity to hyper parameters: aspect ratio

settings	anchor scales	aspect ratios	mAP (%)
1 scale, 1 ratio	128^2	1:1	65.8
	256^2	1:1	66.7
1 scale, 3 ratios	128^2	{2:1, 1:1, 1:2}	68.8
	256^2	{2:1, 1:1, 1:2}	67.9
3 scales, 1 ratio	$\{128^2, 256^2, 512^2\}$	1:1	**69.8**
3 scales, 3 ratios	$\{128^2, 256^2, 512^2\}$	{2:1, 1:1, 1:2}	**69.9**

Sensitivity to hyper parameters: lambdas

λ	0.1	1	10	100
mAP (%)	67.2	68.9	69.9	69.1

Sensitivity to hyper parameters: proposals

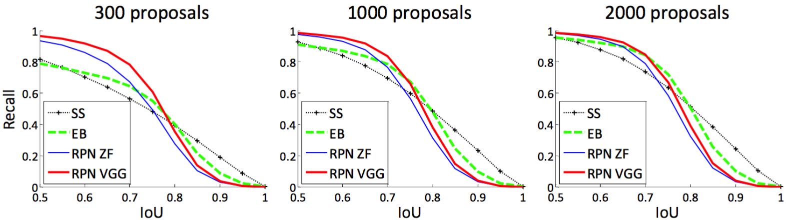

Figure 4: Recall *vs.* IoU overlap ratio on the PASCAL VOC 2007 test set.

MS COCO

method	# box	data	mAP
SS	2000	07	66.9
SS	2000	07+12	70.0
RPN*	300	07	68.5
RPN	300	07	69.9
RPN	300	07+12	73.2
RPN	300	COCO+07+12	**78.8**

Literature overview

- Kislyuk, D., Liu, Y., Liu, D., Tzeng, E., & Jing, Y. (2015). Human Curation and Convnets: Powering Item-to-Item Recommendations on Pinterest. arXiv:1511.04003 [Cs], 16. Retrieved from http://arxiv.org/abs/1511.04003
- Schiele, B., Hosang, J., Benenson, R., & Doll, P. (2016). What Makes for Effective Detection Proposals ?, 38(4), 814830.
- He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. Arxiv.Org, 7(3), 171180. http://doi.org/10.3389/fpsyg.2013.00124

Discussion

- Complicated 4-stage raining pipeline. Open research question: how to incorporate derivatives of proposal coordinates into the objective function.
- Most of conclusion has been obtained experimentally. No theoretical explanation about influences of hyperparameters and alternating training.

Questions?