Review: Fast R-CNN (Object Detection)

SH Tsang Follow
Sep 4, 2018 · 5 min read

In this story, Fast Region-based Convolutional Network method (Fast R-CNN) [1] is reviewed. It improves the training and testing speed as well as increasing the detection accuracy.

1. **Fast R-CNN trains the very deep VGG-16 [2] 9× faster than R-CNN [3], 213× faster at test time**
2. **Higher mAP on PASCAL VOC 2012**
3. **Compared to SPPNet [4], it trains VGG-16 3× faster, tests 10× faster, and is more accurate.**

This is an **2015 ICCV paper** with **over 3000 citations** when I was writing this story. (SH Tsang @ Meidum)

. . .

What are covered

1. **The Problems of Prior Arts**
2. **ROI Pooling Layer**
3. **Multi-task Loss**
4. **Some Other Ablation Study**
5. **Comparison with State-of-the-art Results**

. . .

1. The Problems of Prior Arts

1.1. Multi-stage Pipeline

R-CNN and SPPNet first trains the CNN for softmax classifier, then uses the feature vectors for training the bounding box regressor. Thus, **R-CNN and SPPNet are not end-to-end training.**

1.2. Expensive in Space and Time

As the **feature vectors are stored in harddisk, occupied hundreds of gigabyte**, for training the bounding box regressor.

1.3. Slow Object Detection

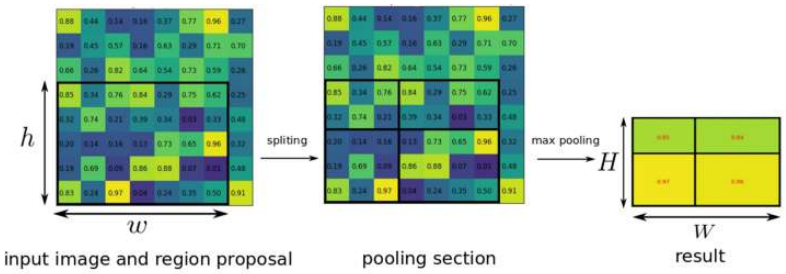
At test-time, R-CNN using VGG-16 needs 47s per image using GPU which is slow.

Fast R-CNN solves above problems!

. . .

2. ROI Pooling Layer

This is actually **a special case of SPP layer in SPPNet with only one pyramid used**. Below illustrates the example:



ROI Pooling

Suppose we got the **region proposal (left) with $h \times w$**, and we would like to have an **output (right) of H×W** sizes of output layer after pooling. Then, the **area for each pooling area (middle)** $= h/H \times w/W$.

And in the example above, with **input ROI of 5×7**, and **output of 2×2**, the **area for each pooling area is 2×3 or 3×3** after rounding.

And the maximum value within the pooling window is taken as output value for each grid which is the same idea of conventional max pooling layer.

. . .

3. Multi-task Loss

Since Fast R-CNN is an end-to-end learning architecture to learn the class of object as well as the associated bounding box position and size, the loss is multi-task loss.

$$L(p, u, t^u, v) = L_{\text{cls}}(p, u) + \lambda[u \geq 1]L_{\text{loc}}(t^u, v)$$

$$L_{\text{cls}}(p, u) = -\log p_u$$

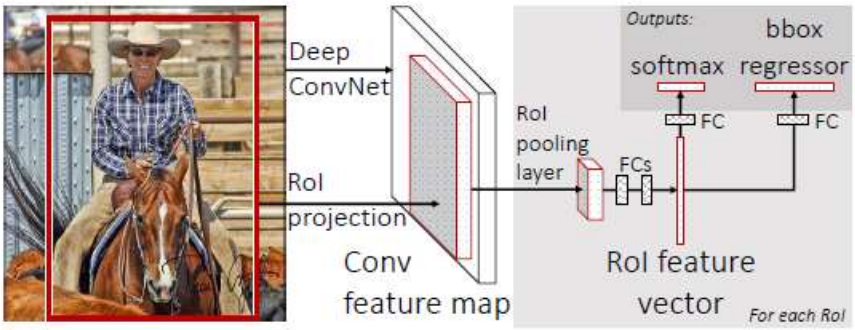
$$L_{\text{loc}}(t^u, v) = \sum_{i \in \{\text{x,y,w,h}\}} \text{smooth}_{L_1}(t_i^u - v_i)$$

$$\text{smooth}_{L_1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases}$$

Multi-task Loss

***L_cls* is the log loss for true class *u*.**
L_los is the loss for bounding box.
[u≥1] means it is equal to 1 when u≥1. (u=0 is background class)

Compared with OverFeat, R-CNN, and SPPNet, Fast R-CNN uses multi-task loss to achieve end-to-end learning.



Fast R-CNN

With mutli-task loss, at the ouput, we have softmax and bounding box regressor as shown at the top right of the figure.

3 Models are evaluated:
S = AlexNet or CaffeNet
M = VGG-like wider version of S
L = VGG-16

	S				M				L			
multi-task training?		✓		✓		✓		✓		✓		✓
stage-wise training?			✓				✓				✓	
test-time bbox reg?			✓	✓			✓	✓			✓	✓
VOC07 mAP	52.2	53.3	54.6	**57.1**	54.7	55.5	56.6	**59.2**	62.6	63.4	64.0	**66.9**

Multi-task Loss Results

With multi-task loss, higher mAP is obtained compared with stage-wise training, i.e. separate training of softmax and bounding box regressor.

. . .

4. Some Other Ablation Study

4.1 Multi Scale Training and Testing

An input image is tested using 5 scales.

	SPPnet ZF		S		M		L
scales	1	5	1	5	1	5	1
test rate (s/im)	0.14	0.38	**0.10**	0.39	0.15	0.64	0.32
VOC07 mAP	58.0	59.2	57.1	58.4	59.2	60.7	**66.9**

1-Scale vs 5-Scale

With 5-scale, higher mAP is obtained for every model with the cost of larger test rate (seconds/image).

4.2 SVM vs Softmax

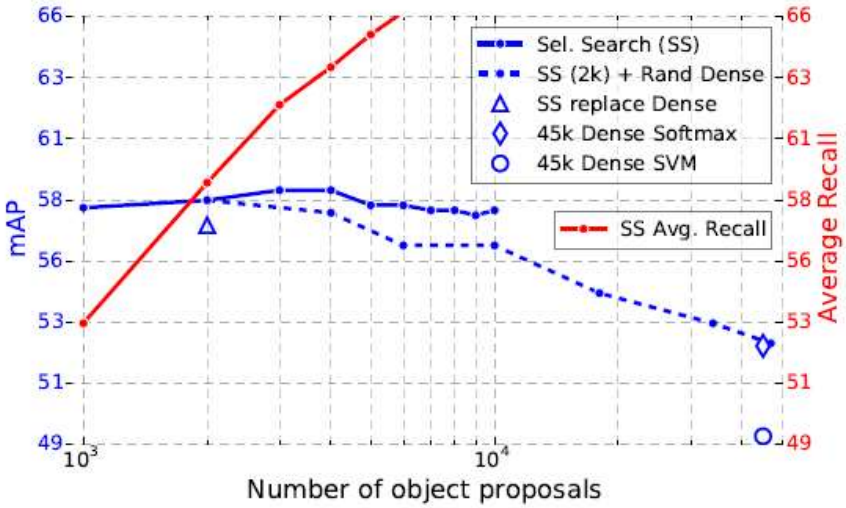
	method	classifier	S	M	L
	R-CNN [9, 10]	SVM	**58.5**	**60.2**	66.0
SVM	FRCN [ours]	SVM	56.3	58.7	66.8
softmax	FRCN [ours]	softmax	57.1	59.2	**66.9**

SVM vs Softmax

In Fast R-CNN (FRCN), **softmax is better than SVM.**

Also, for SVM, the feature vectors need to be stored for hundreds of gigabyte in harddisk, and become stage-wise training while softmax can achieve end-to-end learning without storing feature vectors into harddisk.

4.3 Region Proposals



Different Proposal Approaches

It is found that **increasing number of region proposals does not necessary increase mAP.**

Spare Set using Selective Search (SS) [5] is already good enough as shown in the figure above (Blue solid line) (SS [5] is being used in R-CNN.)

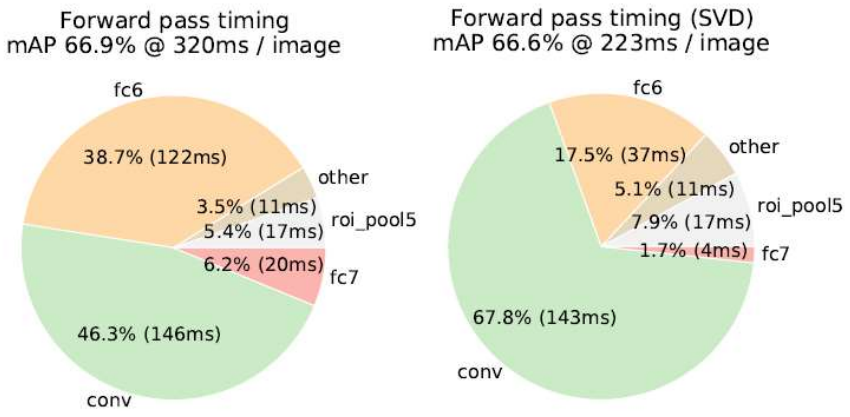
It is still a problem that Fast R-CNN needs region proposals from an external source.

4.4 Truncated SVD for faster detection

One of the bottlenecks of testing time is at FC layers.

Authors use Singular Vector Decomposition (SVD) to reduce the number of connection in order to decrease the test time.

The top 1024 singular values from 25088×4096 matrix in FC6 layer, and **the top 256 singular values from 4096×4096 matrix in FC7 layer.**



Large Reduction of Test Time for FC6 and FC7 Layers

• • •

5. Comparison with State-of-the-art Results

5.1 VOC 2007

method	train set	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	persn	plant	sheep	sofa	train	tv	mAP
SPPnet BB [11]†	07 \ diff	73.9	72.3	62.5	51.5	44.4	74.4	73.0	74.4	42.3	73.6	57.7	70.3	74.6	74.3	54.2	34.0	56.4	56.4	67.9	73.5	63.1
R-CNN BB [10]	07	73.4	77.0	63.4	45.4	**44.6**	75.1	78.1	79.8	40.5	73.7	62.2	79.4	78.1	73.1	64.2	**35.6**	66.8	67.2	70.4	**71.1**	66.0
FRCN [ours]	07	74.5	78.3	69.2	53.2	36.6	77.3	78.2	82.0	40.7	72.7	67.9	79.6	79.2	73.0	69.0	30.1	65.4	70.2	75.8	65.8	66.9
FRCN [ours]	07 \ diff	74.6	**79.0**	68.6	57.0	39.3	79.5	**78.6**	81.9	**48.0**	74.0	67.4	80.5	80.7	74.1	69.6	31.8	67.1	68.4	75.3	65.5	68.1
FRCN [ours]	07+12	**77.0**	78.1	**69.3**	**59.4**	38.3	**81.6**	**78.6**	**86.7**	42.8	**78.8**	**68.9**	**84.7**	**82.0**	**76.6**	**69.9**	31.8	**70.1**	**74.8**	**80.4**	70.4	**70.0**

Fast R-CNN

Fast R-CNN with difficult examples removed

Fast R-CNN with external data training

VOC 2007 Results

Fast R-CNN: 66.9% mAP
Fast R-CNN with difficult examples removed during training (This is the setting of SPPNet): 68.1% mAP
Fast R-CNN with external VOC 2012 trained: 70.0% mAP

5.2 VOC 2010

method	train set	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	persn	plant	sheep	sofa	train	tv	mAP
BabyLearning	Prop.	77.7	73.8	62.3	48.8	45.4	67.3	67.0	80.3	41.3	70.8	49.7	79.5	74.7	78.6	64.5	36.0	69.9	55.7	70.4	61.7	63.8
R-CNN BB [10]	12	79.3	72.4	63.1	44.0	44.4	64.6	66.3	84.9	38.8	67.3	48.4	82.3	75.0	76.7	65.7	35.8	66.2	54.8	69.1	58.8	62.9
SegDeepM	12+seg	**82.3**	75.2	67.1	50.7	**49.8**	71.1	69.6	88.2	42.5	71.2	50.0	85.7	76.6	81.8	69.3	**41.5**	**71.9**	62.2	73.2	**64.6**	67.2
FRCN [ours]	12	80.1	74.4	67.7	49.4	41.4	74.2	68.8	87.8	41.9	70.1	50.2	86.1	77.3	81.1	70.4	33.3	67.0	63.3	77.2	60.0	66.1
FRCN [ours]	07++12	82.0	**77.8**	**71.6**	**55.3**	42.4	**77.3**	**71.7**	**89.3**	**44.5**	**72.1**	**53.7**	**87.7**	**80.0**	**82.5**	**72.7**	36.6	68.7	**65.4**	**81.1**	62.7	**68.8**

VOC 2010 Results

Similar to VOC 2007, **Fast R-CNN with external VOC 2007 and 2012 trained is the best with 68.8% mAP.**

5.3 VOC 2012

method	train set	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	persn	plant	sheep	sofa	train	tv	mAP
BabyLearning	Prop.	78.0	74.2	61.3	45.7	42.7	68.2	66.8	80.2	40.6	70.0	49.8	79.0	74.5	77.9	64.0	35.3	67.9	55.7	68.7	62.6	63.2
NUS_NIN_c2000	Unk.	80.2	73.8	61.9	43.7	**43.0**	70.3	67.6	80.7	41.9	69.7	51.7	78.2	75.2	76.9	65.1	**38.6**	**68.3**	58.0	68.7	63.3	63.8
R-CNN BB [10]	12	79.6	72.7	61.9	41.2	41.9	65.9	66.4	84.6	38.5	67.2	46.7	82.0	74.8	76.0	65.2	35.6	65.4	54.2	67.4	60.3	62.4
FRCN [ours]	12	80.3	74.7	66.9	46.9	37.7	73.9	68.6	87.7	41.7	71.1	51.1	86.0	77.8	79.8	69.8	32.1	65.5	63.8	76.4	61.7	65.7
FRCN [ours]	07++12	**82.3**	**78.4**	**70.8**	**52.3**	38.7	**77.8**	**71.6**	**89.3**	**44.2**	**73.0**	**55.0**	**87.5**	**80.5**	**80.8**	**72.0**	35.1	**68.3**	**65.7**	**80.4**	**64.2**	**68.4**

VOC 2012 Results

Similar to VOC 2007, **Fast R-CNN with external VOC 2007 trained is the best with 68.4% mAP.**

5.4 Training and Testing Time

	Fast R-CNN			R-CNN			SPPnet
	S	**M**	**L**	**S**	**M**	**L**	†**L**
train time (h)	**1.2**	2.0	9.5	22	28	84	25
train speedup	**18.3**×	14.0×	8.8×	1×	1×	1×	3.4×
test rate (s/im)	0.10	0.15	0.32	9.8	12.1	47.0	2.3
▷ with SVD	**0.06**	0.08	0.22	-	-	-	-
test speedup	98×	80×	146×	1×	1×	1×	20×
▷ with SVD	169×	150×	**213**×	-	-	-	-
VOC07 mAP	57.1	59.2	**66.9**	58.5	60.2	66.0	63.1
▷ with SVD	56.5	58.7	66.6	-	-	-	-

Training and Testing Time

As mentioned, **Fast R-CNN trains the very deep VGG-16 [2] 9× faster than R-CNN [3], 213× faster at test time.**

Compared to SPPNet [4], it trains VGG-16 3× faster, and tests 10× faster.

. . .

References

1. [2015 ICCV] [Fast R-CNN]
   Fast R-CNN
2. [2015 ICLR] [VGGNet]
   Very Deep Convolutional Networks for Large-Scale Image Recognition
3. [2014 CVPR] [R-CNN]
   Rich feature hierarchies for accurate object detection and semantic segmentation
4. [2014 ECCV] [SPPNet]
   Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition
5. [2013 IJCV] [Selective Search]
   Selective Search for Object Recognition

My Reviews

1. Review: R-CNN (Object Detection)
2. Review of AlexNet, CaffeNet—Winner of ILSVRC 2012 (Image Classification)
3. Review: SPPNet—1st Runner Up (Object Detection), 2nd Runner Up (Image Classification) in ILSVRC 2014
4. Review: VGGNet—1st Runner-Up (Image Classification), Winner (Localization) in ILSVRC 2014
5. Review: OverFeat—Winner of ILSVRC 2013 Localization Task (Object Detection)