Deep Feature Pyramid Reconfiguration for Object Detection

Tao Kong[1], Fuchun Sun[1], Wenbing Huang[2], Huaping Liu[1]

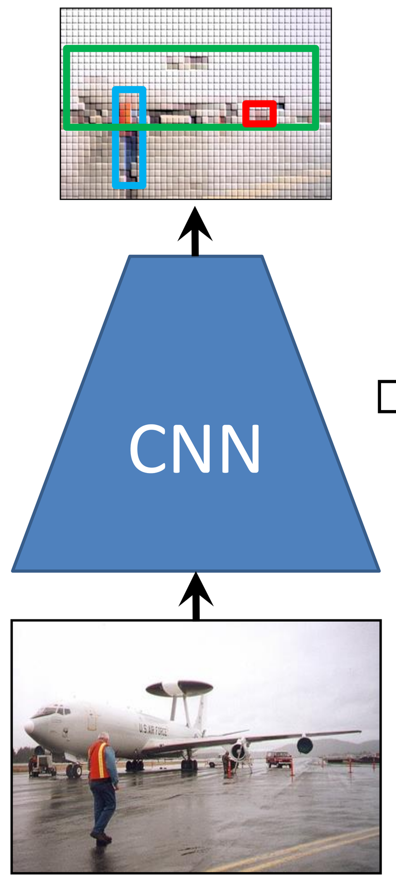
[1]Department of CST, Tsinghua University, [2]Tencent AI Lab

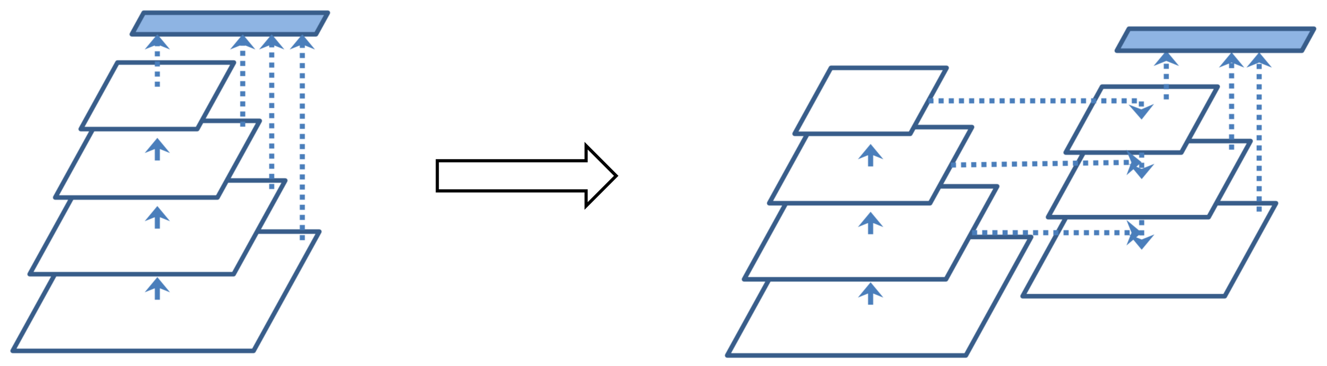
ECCV 2018 European Conference on Computer Vision

8 – 14 September 2018 | Munich, Germany

Feature pyramid based object detectors

Key idea:
Detecting objects of multiple scales at corresponding feature levels





SSD,
Liu, W, et al. *ECCV*, 2016.
MSCNN,
Cai, Z, et al. *ECCV*, 2016.

FPN, Lin, S, et al. CVPR 2017.
RON, Kong, T, et al. CVPR2017
RetinaNet, Lin, S, et al. ICCV, 2017.

Take a deeper look at FPN

Small objects,
High resolution
Low semantics

Large objects,
Low resolution
High semantics

The total backbone network outputs: $X_{net} = \{x_1, x_2, ..., x_L\}$,

In SSD the prediction feature map sets can be expressed as: $X_{pred} = \{x_P, x_{P+1}, \ldots, x_L\}$

In FPN, we get

$$
\begin{aligned}
x'_L &= x_L, \\
x'_{L-1} &= \alpha_{L-1} \cdot x_{L-1} + \beta_{L-1} \cdot x_L, \\
x'_{L-2} &= \alpha_{L-2} \cdot x_{L-2} + \beta_{L-2} \cdot x'_{L-1}, \\
&= \alpha_{L-2} \cdot x_{L-2} + \beta_{L-2}\alpha_{L-1} \cdot x_{L-1} + \beta_{L-2}\beta_{L-1} \cdot x_L,
\end{aligned}
$$

$$x'_l = \sum_{l=P}^{L} w_l \cdot x_l.$$

linear combination

$$X'_{pred} = \{x'_P, x'_{P+1}, \ldots, x'_L\}.$$

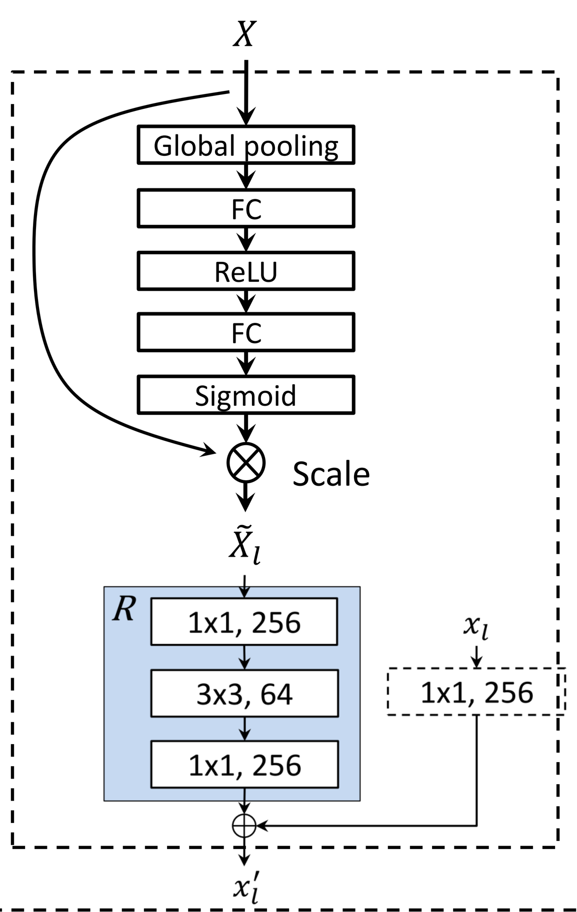
Deep Feature Reconfiguration

Feature generating process at *l-th* level

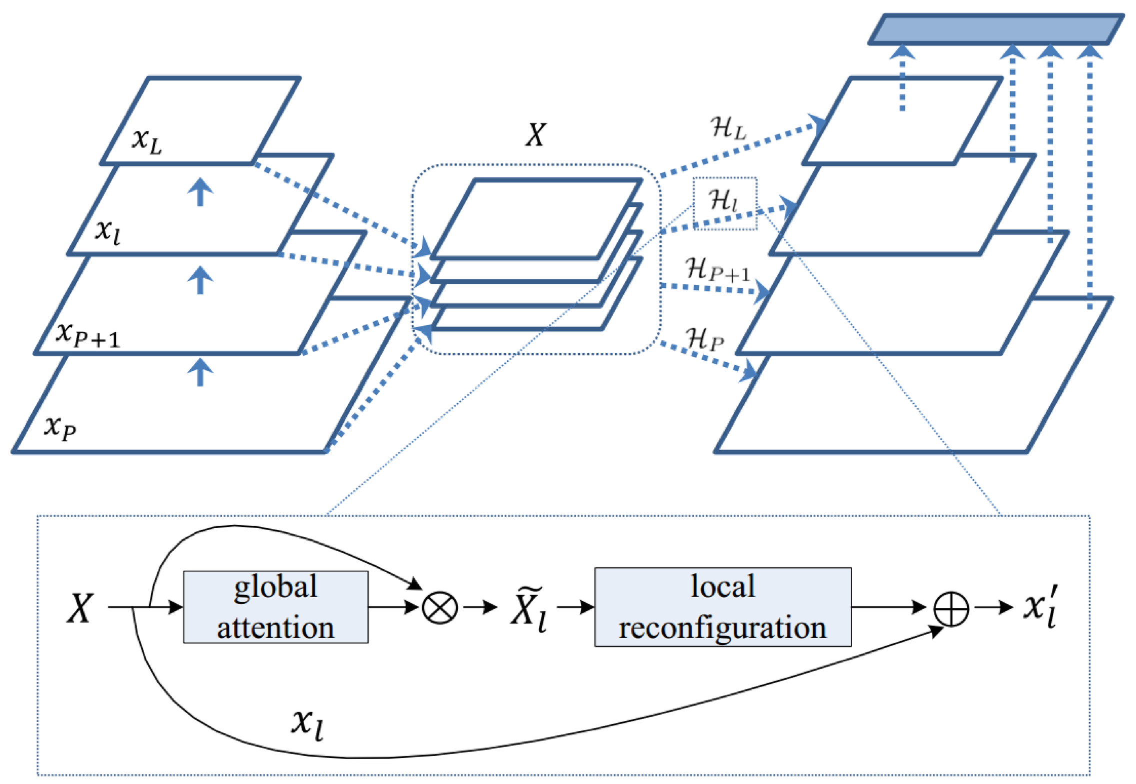
$$x'_l = \mathcal{H}_l(X)$$

feature hierarchy

non-linear transformation



Methodology



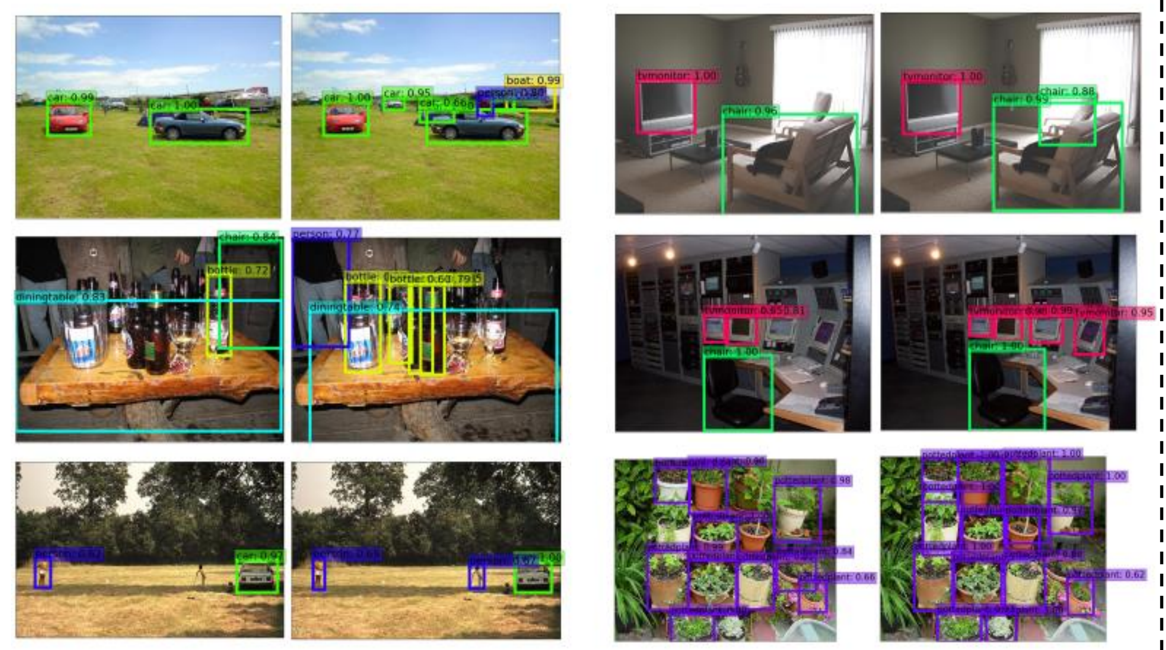
Advantages

- ✓ The deeper layers also have more opportunities to re-organize its features, and has more potential for boosting results;
- ✓ The global attention makes the network to focus more on features with suitable semantics;
- ✓ The local residual learn block gives more opportunity to better model the feature hierarchy.

Main results

method	train Data	input size	network	Average Precision 0.5	0.75	0.5:0.95
two-stage						
OHEM++[43]	trainval	$\sim 1000 \times 600$	VGG-16	45.9	26.1	25.5
Faster[39]	trainval	$\sim 1000 \times 600$	VGG-16	42.7	-	21.9
R-FCN[6]	trainval	$\sim 1000 \times 600$	ResNet-101	51.9	-	29.9
CoupleNet[49]	trainval35k	$\sim 1000 \times 600$	ResNet-101	**54.8**	37.2	34.4
one-stage						
SSD300[34]	trainval35k	300×300	VGG-16	43.1	25.8	25.1
SSD512[34]	trainval35k	512×512	VGG-16	48.5	30.3	28.8
SSD513[15]	trainval35k	513×513	ResNet-101	50.4	33.1	31.2
DSSD321[15]	trainval35k	321×321	ResNet-101	46.1	29.2	28.0
DSSD513[15]	trainval35k	513×513	ResNet-101	53.3	35.2	33.2
RON320[26]	trainval	320×320	VGG-16	47.5	25.9	26.2
YOLOv2[38]	trainval35k	544×544	DarkNet-19	44.0	19.2	21.6
RetinaNet[31]	trainval35k	500×500	ResNet-101	53.1	36.8	34.4
Ours300	trainval	300×300	VGG-16	48.2	29.1	28.4
Ours512	trainval	512×512	VGG-16	50.9	32.2	31.5
Ours300	trainval	300×300	ResNet-101	50.5	32.0	31.3
Ours512	trainval	512×512	ResNet-101	54.3	**37.3**	**34.6**

MS COCO test-dev2015 detection results.



SSD300 Ours300 SSD300 Ours300

method	backbone	FPS	mAP(%)
SSD (Caffe) [34]	VGG-16	46	77.5
SSD (ours-re)	VGG-16	44	77.5
SSD+lateral	VGG-16	37	78.5
SSD+Local only	VGG-16	40	79.0
SSD+Local only(no res)	VGG-16	40	78.6
SSD+Global-Local	VGG-16	39.5	**79.6**

Effectiveness of designs within SSD (VOC 2007 Test)

method	backbone	mAP(%)
Faster [39]	VGG-16	73.2
Faster [6]	ResNet-101	76.4
Faster(ours-re)	ResNet-50	77.6
Faster(ours-re)	ResNet-101	78.9
Faster+FPNs	ResNet-50	78.8
Faster+FPNs	ResNet-101	79.8
Faster+Global-Local	ResNet-50	79.4
Faster+Global-Local	ResNet-101	**80.6**

Effectiveness of designs
within Faster R-CNN (VOC 2007 Test)

Kong T, Sun F, Huang W, et al. Deep Feature Pyramid Reconfiguration for Object Detection[J]. arXiv preprint arXiv:1808.07993, 2018.

I expect to graduate in July 2019, for more information: https://taokong.github.io