Object Detection based on Region Decomposition and Assembly

Seung-Hwan Bae
Computer Vision Lab., Department of Computer Science and Engineering
Incheon National University, 119 Academy-ro, Yeonsu-gu, Incheon, 22012, Korea
shbae@inu.ac.kr

Author

Seung-Hwan Bae

Assistant Professor, Department of Computer Science and Engineering, Incheon National University
在 inu.ac.kr 的电子邮件经过验证 - 首页

Computer Vision Machine Learning Tracking

标题	引用次数	年份
Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning SH Bae, KJ Yoon Proceedings of the IEEE conference on computer vision and pattern …	307	2014
Confidence-based data association and discriminative deep appearance learning for robust online multi-object tracking SH Bae, KJ Yoon IEEE transactions on pattern analysis and machine intelligence 40 (3), 595-610	93	2017
Robust Online Multi-Object Tracking with Data Association and Track Management SH Bae, KJ Yoon Image Processing, IEEE Transactions on 23 (7), 2820-2833	48	2014
Polyp detection via imbalanced learning and discriminative feature learning SH Bae, KJ Yoon IEEE transactions on medical imaging 34 (11), 2379-2393	41	2015
Joint initialization and tracking of multiple moving objects using Doppler information JH Yoon, DY Kim, SH Bae, V Shin IEEE Transactions on Signal Processing 59 (7), 3447-3452	36	2011
Learning discriminative appearance models for online multi-object tracking with appearance discriminability measures SH Lee, MY Kim, SH Bae IEEE Access 6, 67316-67328	10	2018

About Object Detection

▶ Region-based object detection infers object regions for one or more categories in an image.

▶Object detectors based on convolutional neural networks (CNNs) are flourishing.

▶ The most notable work is the R-CNN (Girshick et al. 2014) framework, followed by Fast R-CNN(Girshick 2015).

▶ As a result, the detection accuracy can be also enhanced by joint learning of region proposal and classification modules.

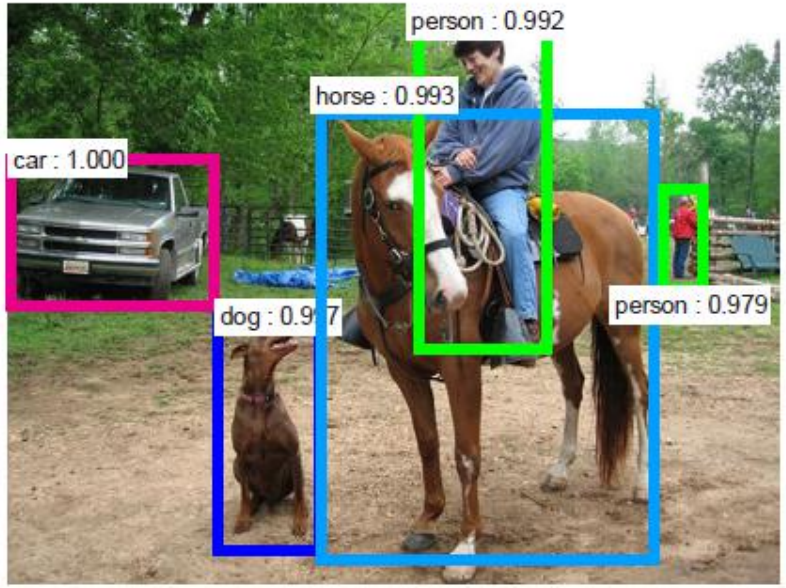
Recent works' disadvantages

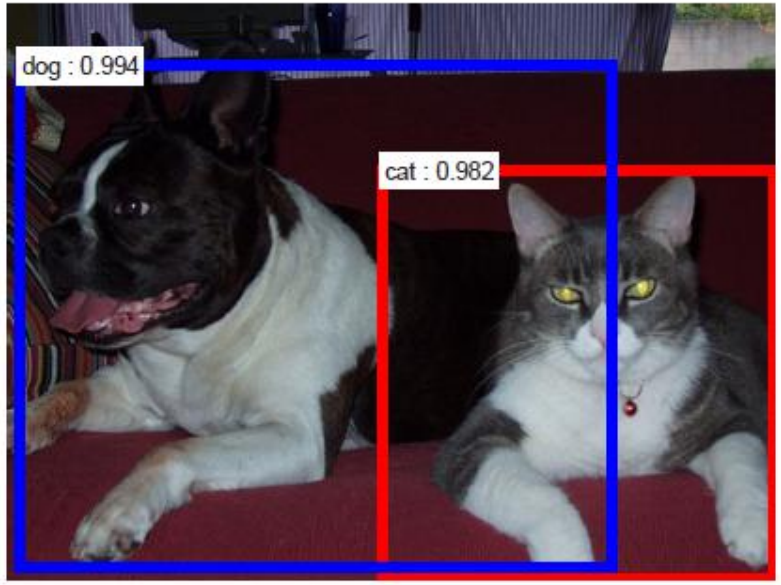
▶ But even with the robustness of the CNNs to the scale variance, the region proposal accuracy is frequently degraded by the mismatches of produced proposals and object regions.

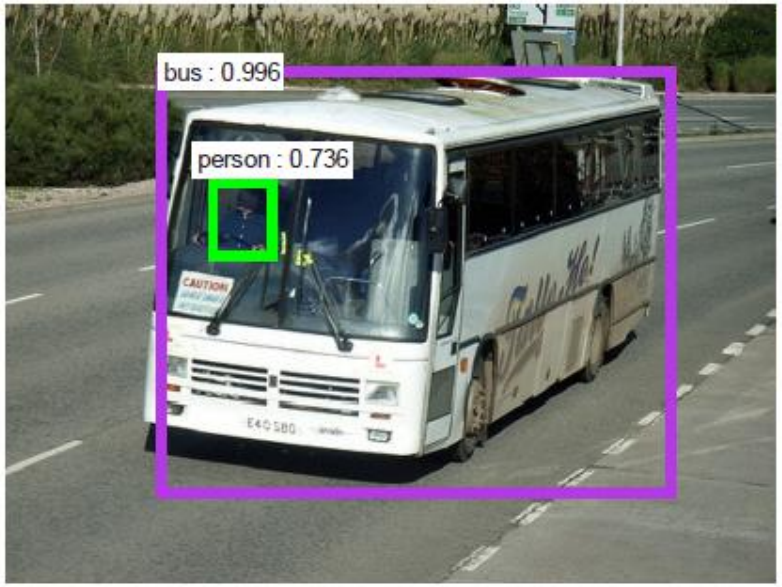
▶ The mismatch tends to be increased for the small object detection.

▶ To improve the proposal accuracy, multi-scale feature representation using feature pyramid is used for generating stronger synthetic feature maps. However, featurizing each level of an image pyramid increases the inference time significantly.

Introduction







Contributions

In this paper, we propose a novel region decomposition and assembly detector (R-DAD) to resolve the limitations of the previous methods.

▶ Multi-scale-based region proposal to improve region proposal accuracy of the region proposal network (RPN).

▶ Achievement of state-of-the-art results without employing other performance improvement methods (e.g. feature pyramid, multi-scale testing, data augmentation, model ensemble, etc.) for several detection benchmark challenge on PASCAL07 (mAP of 81.2%), PASCAL12 (mAP of 82.0%), and MSCOCO18 (mAP of 43.1%)

Related work

▶For feature extraction and object classification, the recent object detectors are therefore constructed based on the deep CNNs (Simonyan and Zisserman 2014; He et al. 2016) trained beforehand with large image datasets.

▶ R-CNN (Girshick et al. 2014) first generate object region proposals using the selective search (Uijlings et al. 2013), extract CNN features (Krizhevsky, Sutskever, and Hinton 2012) of the regions, and classify them with class-specific SVMs.

▶Fast RCNN (Girshick 2015) improve the R-CNN speed using feature sharing and RoI pooling.

▶Among several works, Faster-RCNN (Ren et al. 2015) achieve the noticeable performance improvement by integrating RPN and Fast RCNN (Girshick 2015).

▶ Basically, the previous works based on multiple feature maps focus on:
(1) multi-region representation to improving the feature discriminability and diversity.
(2) multi-scale representation to detect the objects with small sizes without image pyramid.

▶ R-DAD can efficiently handle both issues together.

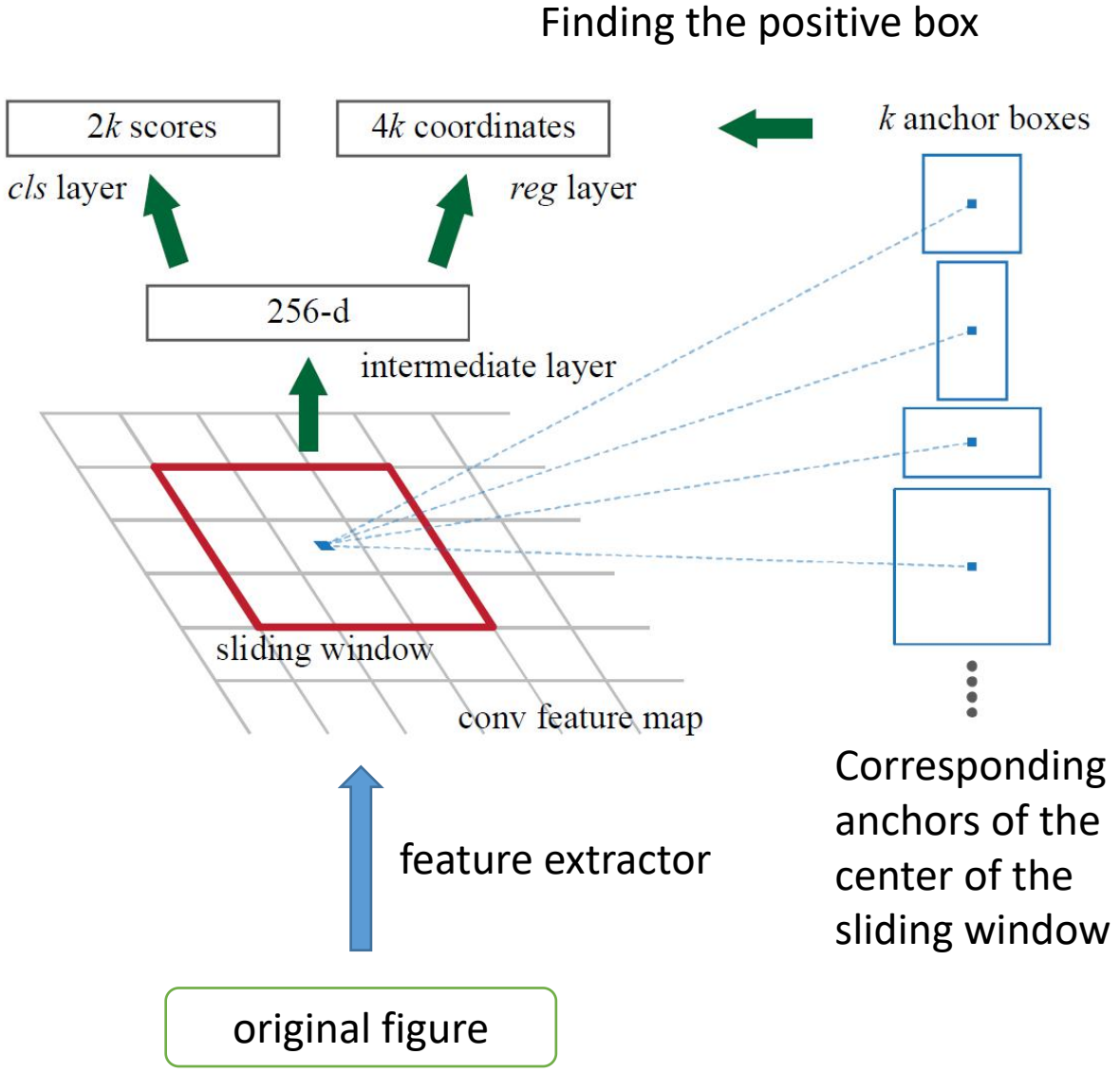
Faster R-CNN (Ren et al. 2015)

▶In the first stage, an input image is resized to be fixed and is fed into a feature extractor (i.e. pretrained classification.

▶ Then, the PRN uses mid-level features at some selected intermediate level (e.g. “conv4” and “conv5” for VGG and ResNet) for generating class-agnostic box proposals and their confidence scores.

▶In the second stage, features of box proposals are cropped by RoI pooling from the same intermediate feature maps used for box proposals.

▶Then, the features for box proposals are subsequently propagated in other higher layers (e.g. “fc6” followed by “fc7”) to predict a class and refine states (i.e. locations and sizes) for each candidate box.

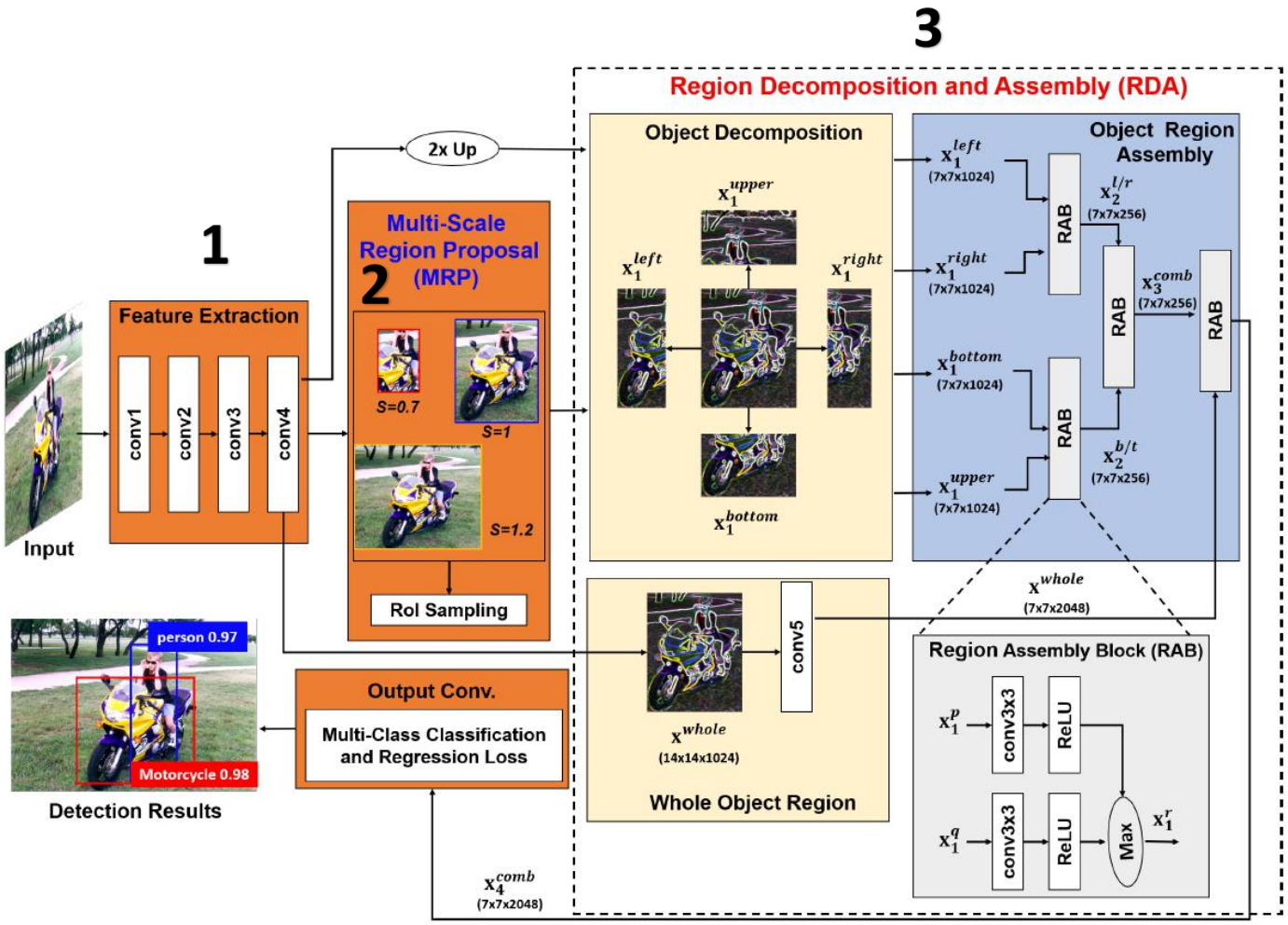


Region Decomposition and Assembly Detector Network Structure Overview

▶ The network mainly consists of feature extraction, multi-scale-based region proposal (MRP) and object region decomposition and assembly (RDA) phases.

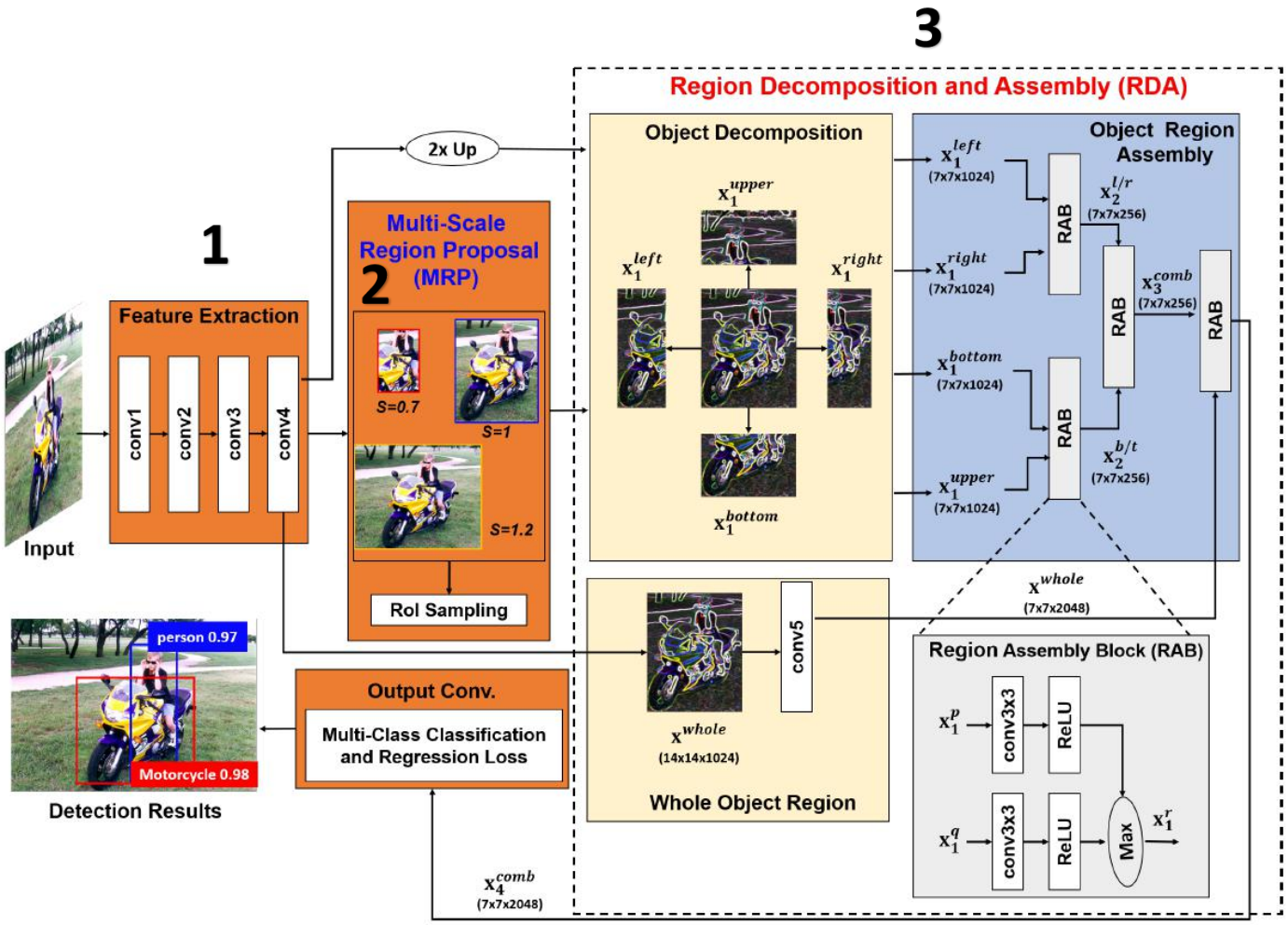
▶1, For extracting a generic CNN features, similar to other works, we use a classification network trained with ImageNet, including ZF-Net (Zeiler and Fergus 2014), VGG16/VGGM1024-Nets (Simonyan and Zisserman 2014), Res-Net101/152 (He et al. 2016)

▶2, In the MRP network, we generate region proposals (i.e. bounding boxes) of different sizes.



Region Decomposition and Assembly Detector Network Structure Overview

▶ 3, We learn the global (i.e. an entire region) and part appearance (i.e. decomposed regions) models in the RDA network. The main process is that we decompose an entire object region into several small regions and extract features of each region.

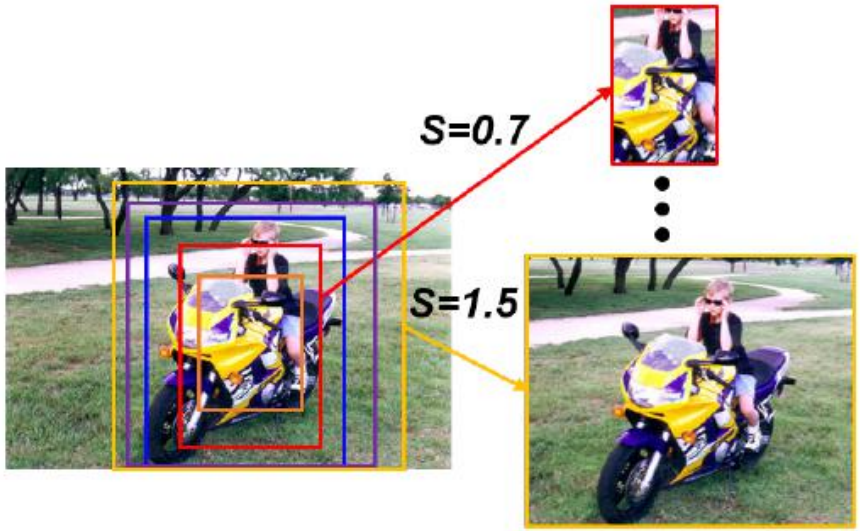


Multi-scale region proposal (MRP) network

▶Each bounding box can be denoted as d = (x, y, w, h), where x, y, w and h are the center positions, width and height. Given a region proposal d, the rescaled box is ds = (x, y, w·s, h·s) with a scaling factor s (≥0). By applying different s to the original box, we can generate ds with different sizes. we use different s = [0.5; 0.7; 1; 1.2; 1.5].

▶ Since huge number of proposals (63×38×9×5) are generated for the feature maps of size 63×38 at the “conv4” layer when using 9 anchors and 5 scale factors.

▶ We maintain the appropriate number of proposals (e.g. 256) by removing the proposals with low confidence and low overlap ratios over ground truth. We then make a ratio of object and non-object samples in a mini-batch to be equal to tackle the samples imbalance problem.



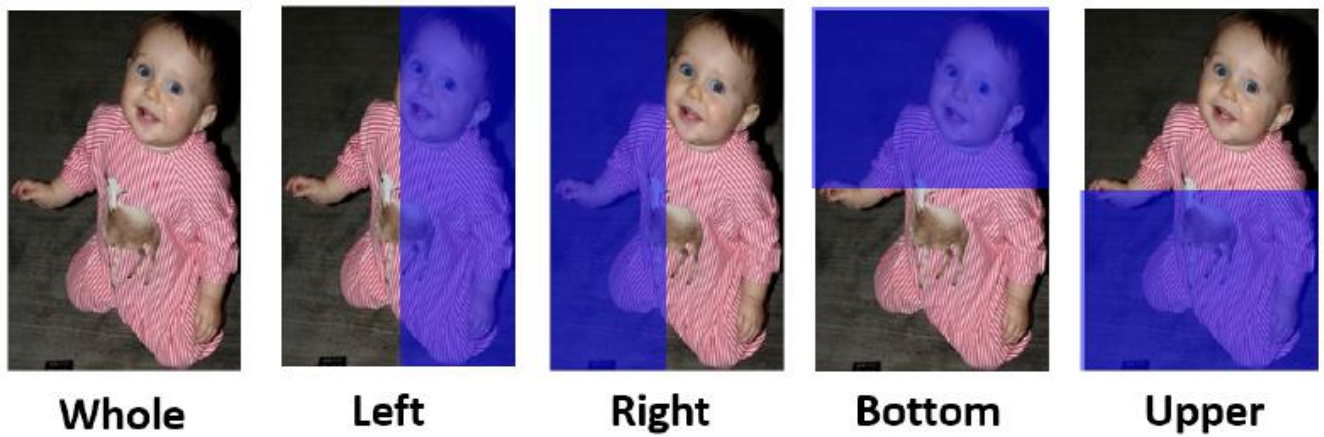


Figure 2: (Left) Rescaled proposals by the MRP. (Right) Several decomposed regions for a whole object region.

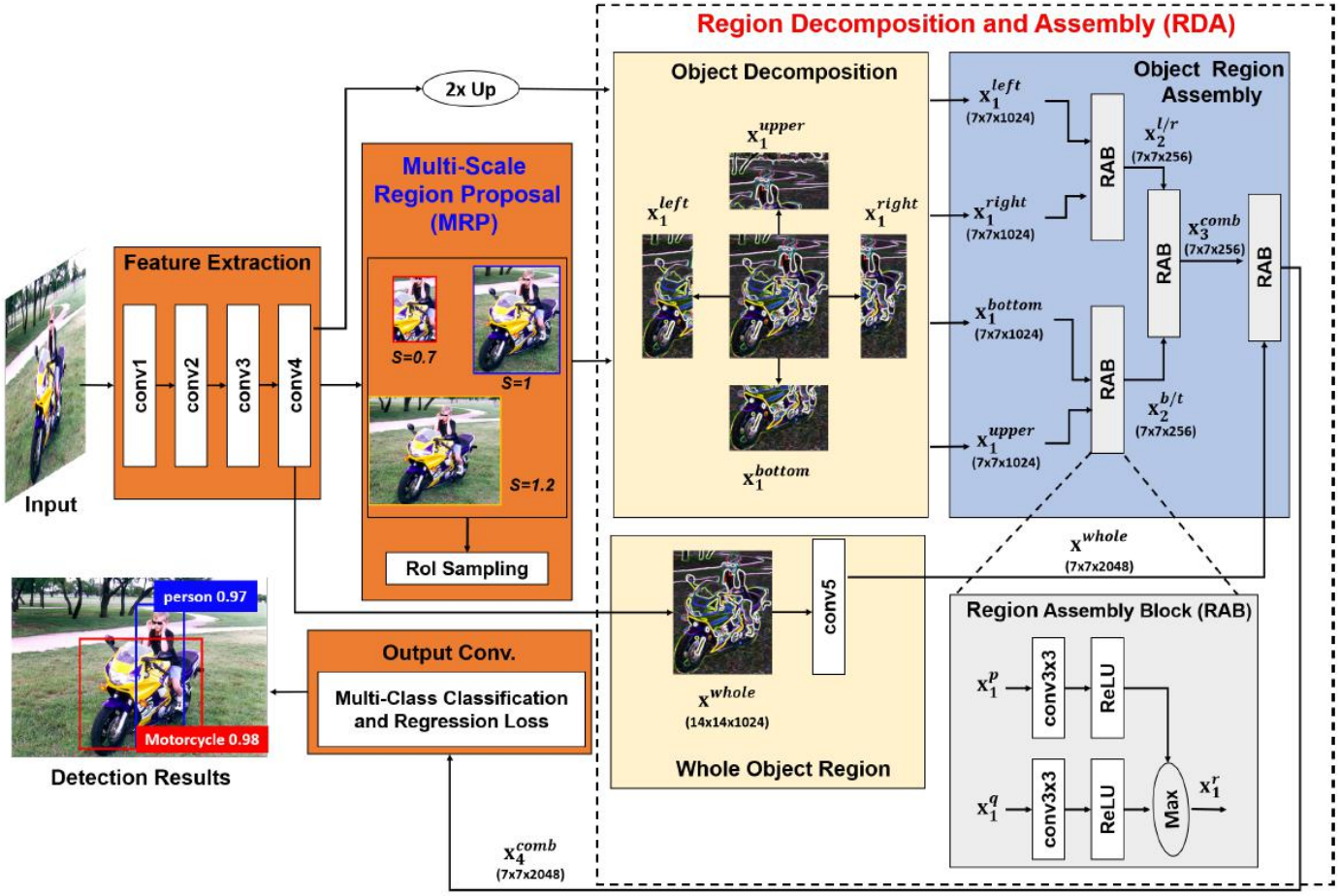
Region decomposition and assembly (RDA) network

Table 1. Using RoI pooling, we also extract warped feature maps of size $\lceil h_{roi}/2 \rceil \times \lceil w_{roi}/2 \rceil$, and denote them as $\mathbf{x}_l^p, p \in \{\text{left, right, bottom, upper}\}$.

In the forward propagation, we convolve part features $\mathbf{x}_{i,l-1}^p$ at layer $l-1$ of size $h_{l-1}^p \times w_{l-1}^p$ with different kernels $\mathbf{w}_{ij}^l$ of size $m_l \times m_l$, and then pass the convolved features a nonlinear activation function $f(\cdot)$ to obtain an updated feature map $\mathbf{x}_{j,l}^p$ of size $(h_{l-1}^p - m_l + 1) \times (w_{l-1}^p - m_l + 1)$ as

$$\mathbf{x}_{j,l}^p = f\left(\sum_{i=1}^{k_l} \mathbf{x}_{i,l-1}^p * \mathbf{w}_{ij}^l + b_j^l\right), \text{ l= 2, 3, 4} \qquad (1)$$

where p represent each part (left, right, bottom, upper) or combined parts (left-right(l/r), bottom-upper (b/u) and comb) as in Fig. 1. b_j^l is a bias factor, k_l is the number of kernels. $*$ means convolution operation. We use the element-wise ReLU function as $f(\cdot)$ for scaling the linear input.



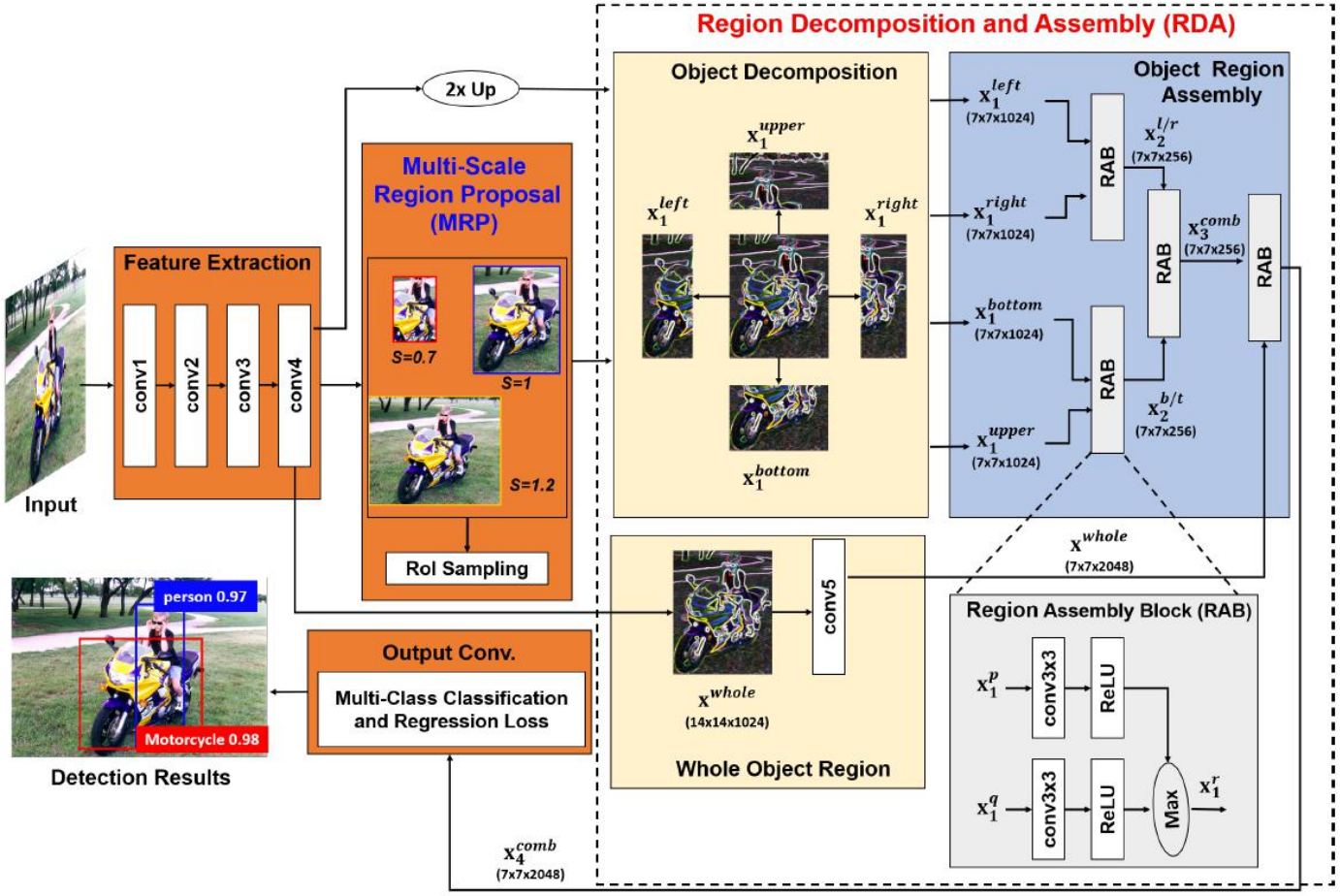
Region decomposition and assembly (RDA) network

Then, the bi-directional outputs $\mathbf{x}_l^p$ and $\mathbf{x}_l^q$ Eq. (1) of different regions are merged to produce the combined feature $\mathbf{x}_l^r$ by using an element-wise max unit over each channel as

$$\mathbf{x}_l^r = \max\left(\mathbf{x}_l^p, \mathbf{x}_l^q\right) \tag{2}$$

p, q and r also represent each part or a combined part as shown in Fig. 2. The element-wise max unit is used to merge information between $\mathbf{x}_l^p$ and $\mathbf{x}_l^q$ and produce $\mathbf{x}_l^r$ with the same size. As a result, the bottom-up feature maps are re-

the last layer is also compared with the combined feature $\mathbf{x}_3^{comb}$ of part models, and then the refined features $\mathbf{x}_4^{comb}$ are connected with the object classification and box regression layers with $cls + 1$ neurons and $4(cls + 1)$ neurons, where cls is the number of object classes and the one is added due to the background class.



R-DAD Training

For each box $\mathbf{d}$, we find the best matched ground truth box $\mathbf{d}^*$ by evaluating IoU. If a box $\mathbf{d}$ has an IoU than 0.5 with any $\mathbf{d}^*$, we assign positive label $o^* \in \{1...cls\}$, and a vector representing the 4 parameterized coordinates of $\mathbf{d}^*$. We assign a negative label (0) to $\mathbf{d}$ that has an IoU between 0.1 and 0.5. From the output layers of the R-DAD, 4 parameterized coordinates and the class label $\hat{o}$ are predicted to each box $\mathbf{d}$. The adjusted box $\hat{\mathbf{d}}$ is generated by applying the predicted regression parameters. For box regression, we use the following parameterization (Girshick et al. 2014).

$$t_x = (\hat{x} - x)/w, \quad t_y = (\hat{y} - y)/h, \\ t_w = \log(\hat{w}/w), \quad t_h = \log(\hat{h}/h), \tag{3}$$

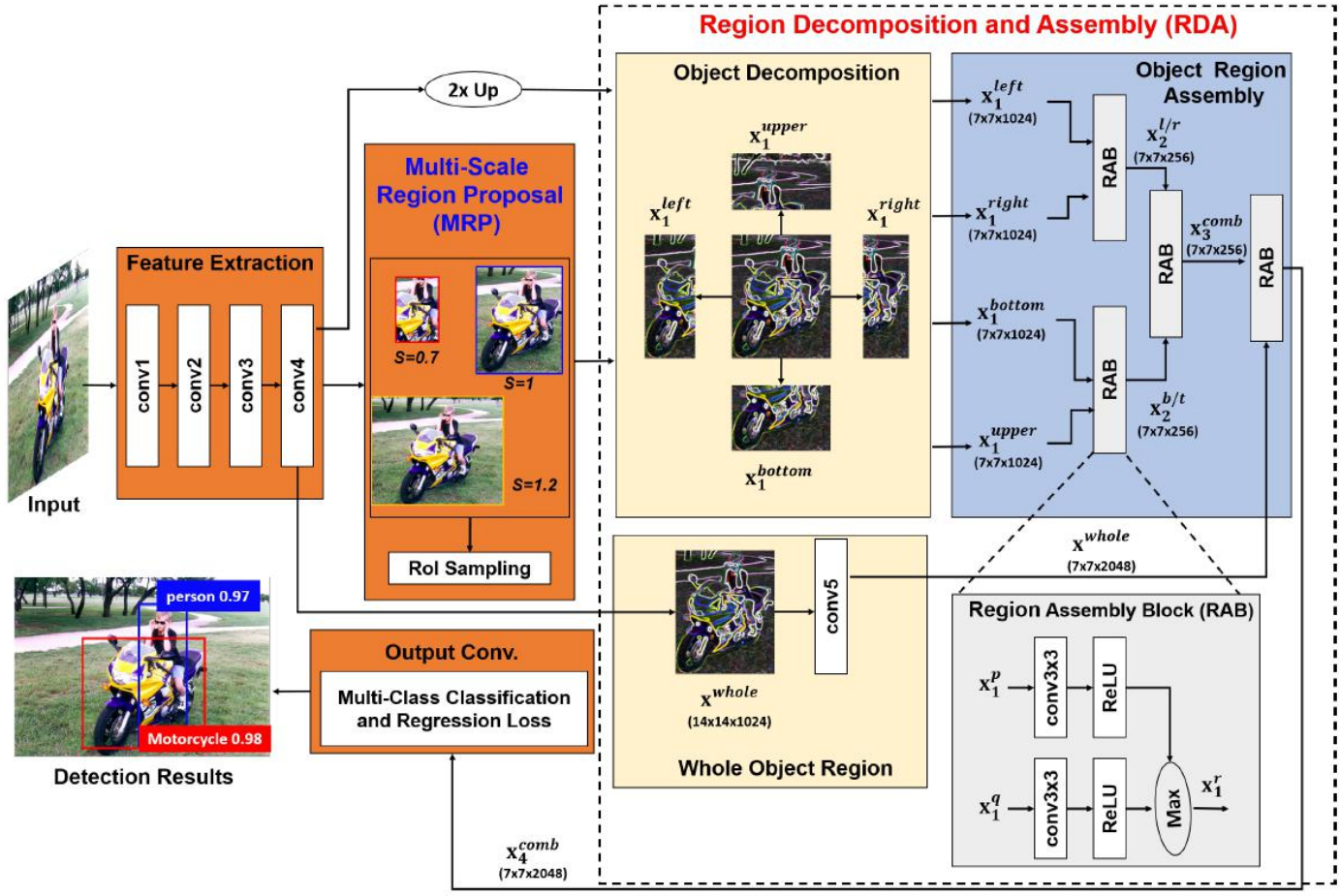
where $\hat{x}$ and x are for the predicted and anchor box, respectively (likewise for y, w, h). Similarly, $\mathbf{t}^* = [t_x^*, t_y^*, t_w^*, t_h^*]$ is evaluated with the predicted box and ground truth boxes. We then train the R-DAD by minimizing the classification and regression losses Eq.(4).

$$L(\mathbf{o}, \mathbf{o}^*, \mathbf{t}, \mathbf{t}^*) = L_{cls}(\mathbf{o}, \mathbf{o}^*) + \lambda [o \geq 1] L_{reg}(\mathbf{t}, \mathbf{t}^*) \tag{4}$$

$$L_{cls}(\mathbf{o}, \mathbf{o}^*) = -\textstyle\sum_u \delta(u, o^*) log(p_u), \tag{5}$$

$$L_{loc}(\mathbf{t}, \mathbf{t}^*) = -\textstyle\sum_{v \in \{x,y,w,h\}} \text{smooth}_{L_1}(t_v, t_v^*), \tag{6}$$

$$\text{smooth}_{L_1}(z) = \begin{cases} 0.5z^2 & \text{if } |z| \leq 1 \\ |z| - 0.5 & \text{otherwise} \end{cases} \tag{7}$$



In Faster R-CNN:

For regression, we adopt the parameterizations of the 4 coordinates following [6]:

$$t_x = (x - x_a)/w_a, \quad t_y = (y - y_a)/h_a, \quad t_w = \log(w/w_a), \quad t_h = \log(h/h_a), \\ t_x^* = (x^* - x_a)/w_a, \quad t_y^* = (y^* - y_a)/h_a, \quad t_w^* = \log(w^*/w_a), \quad t_h^* = \log(h^*/h_a),$$

where x, y, w, and h denote the two coordinates of the box center, width, and height. Variables x, x_a, and x^* are for the predicted box, anchor box, and ground-truth box respectively (likewise for y, w, h). This can be thought of as bounding-box regression from an anchor box to a nearby ground-truth box.

Experimental results

Evaluation measure: We use average precision (AP) per class which is a standard metric for object detection. It is evaluated by computing the area under the precision-recall curve. We also compute mean average precision (mAP) by averaging the APs over all object classes. When evaluating AP and mAP on PASCAL and COCO, we use the public available codes (Girshick 2015; Lin et al. 2014) or evaluation servers for those competition.

Table 2: Ablation study: the detection comparison of different region assembly blocks.

Stage 1	Stage 2	Stage 3	Mean AP
Sum	Sum	Sum	69.61
Sum	Max	Max	69.34
Sum	Sum	Max	69.82
Max	Max	Sum	69.08
Max	Max	Sum $[c_1 = 1, c_2 = \gamma]$	68.80
Max	Max	Sum $[c_1 = \gamma, c_2 = 1]$	69.30
Max	Max	Concatenation	71.95
Max(dil. $d = 4$)	Max	Max	70.87
Max (dil. $d = 2$)	Max(dil. $d = 2$)	Max	70.55
Max	Max(dil. $d = 4$)	Max	70.64
Max($m = 5$)	Max ($m = 5$)	Max	75.10
Max ($m = 3$)	**Max** ($m = 3$)	**Max** ($m = 3$)	**74.90**

Table 1: Ablation study: effects of the proposed multi-scale region proposal and object region decomposition/assembly methods.

Method	Combination						
Multi-scale region proposal $s = [0.7, 1.0, 1.5]$	√			√			
Multi-scale region proposal $s = [0.5, 0.7, 1.0, 1.2, 1.5]$		√			√	√	
Decomposition/assembly			√	√	√	√	
Up-sampling						√	
Mean AP	68.90	70.0	70.30	71.95	72.65	73.90	**74.90**

Sum $[c_1 = \gamma, c_2 = 1]$ means that $\mathbf{x}^{whole}$ and $\mathbf{x}_3^{comb}$ are summed with γ and 1 weights. This is a similar concept to

Moreover, to determine the effective receptive field size, we change the size of convolution kernels with $m = 5$ at the stage 1 and 2 in the RDA network. Moreover, we also try d-dilated convolution filters to expand the receptive field more. However, exploiting the dilated convolutions and 5x5 convolution filters does not increase the mAP significantly.

▶ Comparison with Faster-RCNN

▶Get higher accuracy

Table 3: Comparison between the R-DAD and Faster-RCNN by using different feature extractors on the VOC07 test set.

Train set	Detector	mAP	Train set	Detector	mAP
PASCAL VOC 07	FRCN/ZF	60.8	PASCAL VOC 07++12	FRCN/ZF	66.0
	R-DAD/ZF	**63.7**		**R-DAD/ZF**	**68.2**
	FRCN/VGGM1024	61.0		FRCN/VGGM1024	65.0
	R-DAD/VGGM1024	**65.0**		**R-DAD/VGGM1024**	**69.1**
	FRCN/VGG16	69.9		FRCN/VGG16	73.2
	R-DAD/VGG16	**73.9**		**R-DAD/VGG16**	**78.2**
	FRCN/Res101	74.9		FRCN/Res101	76.6
	R-DAD/Res101	**77.6**		**R-DAD/Res101**	**81.2**

▶Not increase complexity

factors. We found that the spatial sizes of RoI feature maps (h_{roi} and w_{roi}) and convolution filters (m) can affect the speed significantly. When using $h_{roi} = 14$, $w_{roi} = 14$ and $m = 5$ in RABs, R-DAD gets 1.5x $\sim$ 2.1x slower but enhanced only about 0.2% as in Table 2. Therefore, we confirm that adding MRP and RDA networks to the Faster RCNN does not increase the complexity significantly.

▶ Comparison with Faster-RCNN and other benchmarks

Table 4: The speed of the Faster R-CNN (FRCN) and R-DAD (input size: 600 ×1000).

Base Network	**ZF**		**VGGM1024**		**VGG16**		**Res101**			**Res152**		
Detector	FRCN	R-DAD	FRCN	R-DAD	FRCN	R-DAD	FRCN	R-DAD	R-DAD($m = 5$)	FRCN	R-DAD	R-DAD($m = 5$)
Time(sec/frame)	0.041	0.048	0.046	0.054	0.15	0.177	0.208	0.245	0.53	0.301	0.385	0.574

Table 5: Performance comparison with other detectors in PASCAL VOC 2012 challenge. The more results can be found in the PASCAL VOC 2012 website.

Train set	**Detector**	**mAP**	**aero**	**bike**	**bird**	**boat**	**bottle**	**bus**	**car**	**cat**	**chair**	**cow**	**table**	**dog**	**horse**	**mbike**	**person**	**plant**	**sheep**	**sofa**	**train**	**tv**
PASCAL VOC **07++12**	Fast (Girshick 2015)	68.4	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
	Faster (Ren et al. 2015)	70.4	84.9	79.8	74.3	53.9	49.8	77.5	75.9	88.5	45.6	77.1	55.3	86.9	81.7	80.9	79.6	40.1	72.6	60.9	81.2	61.5
	SSD300 (Liu et al. 2016)	72.4	85.6	80.1	70.5	57.6	46.2	79.4	76.1	89.2	53.0	77.0	60.8	87.0	83.1	82.3	79.4	45.9	75.9	69.5	81.9	67.5
	SSD512 (Liu et al. 2016)	74.9	87.4	82.3	75.8	59.0	52.6	81.7	81.5	90.0	55.4	79.0	59.8	88.4	84.3	84.7	83.3	50.2	78.0	66.3	86.3	72.0
	YOLOv2 (Redmon and Farhadi 2017)	73.4	86.3	82.0	74.8	59.2	51.8	79.8	76.5	90.6	52.1	78.2	58.5	89.3	82.5	83.4	81.3	49.1	77.2	62.4	83.8	68.7
	MR-CNN (Gidaris and Komodakis 2015)	73.9	85.5	82.9	76.6	57.8	62.7	79.4	77.2	86.6	55.0	79.1	62.2	87.0	83.4	84.7	78.9	45.3	73.4	65.8	80.3	74.0
	HyperNet (Kong et al. 2016)	71.4	84.2	78.5	73.6	55.6	53.7	78.7	79.8	87.7	49.6	74.9	52.1	86.0	81.7	83.3	81.8	48.6	73.5	59.4	79.9	65.7
	ION (Bell et al. 2016)	76.4	87.5	84.7	76.8	63.8	58.3	82.6	79.0	90.9	57.8	82.0	64.7	88.9	86.5	84.7	82.3	51.4	78.2	69.2	85.2	73.5
	R-DAD/Res101	**80.2**	90.0	86.6	81.3	71.2	66.0	83.4	83.7	94.5	63.2	84.0	64.2	92.8	90.1	88.6	87.3	62.2	82.8	70.9	88.8	72.2
	R-DAD/Res152	**82.0**	90.2	88.1	85.3	73.3	71.4	84.5	87.4	94.6	65.1	86.8	64.0	94.1	89.7	89.2	89.3	64.5	83.5	72.2	89.5	77.6

▶ Comparison of without/with region decomposition assembly method

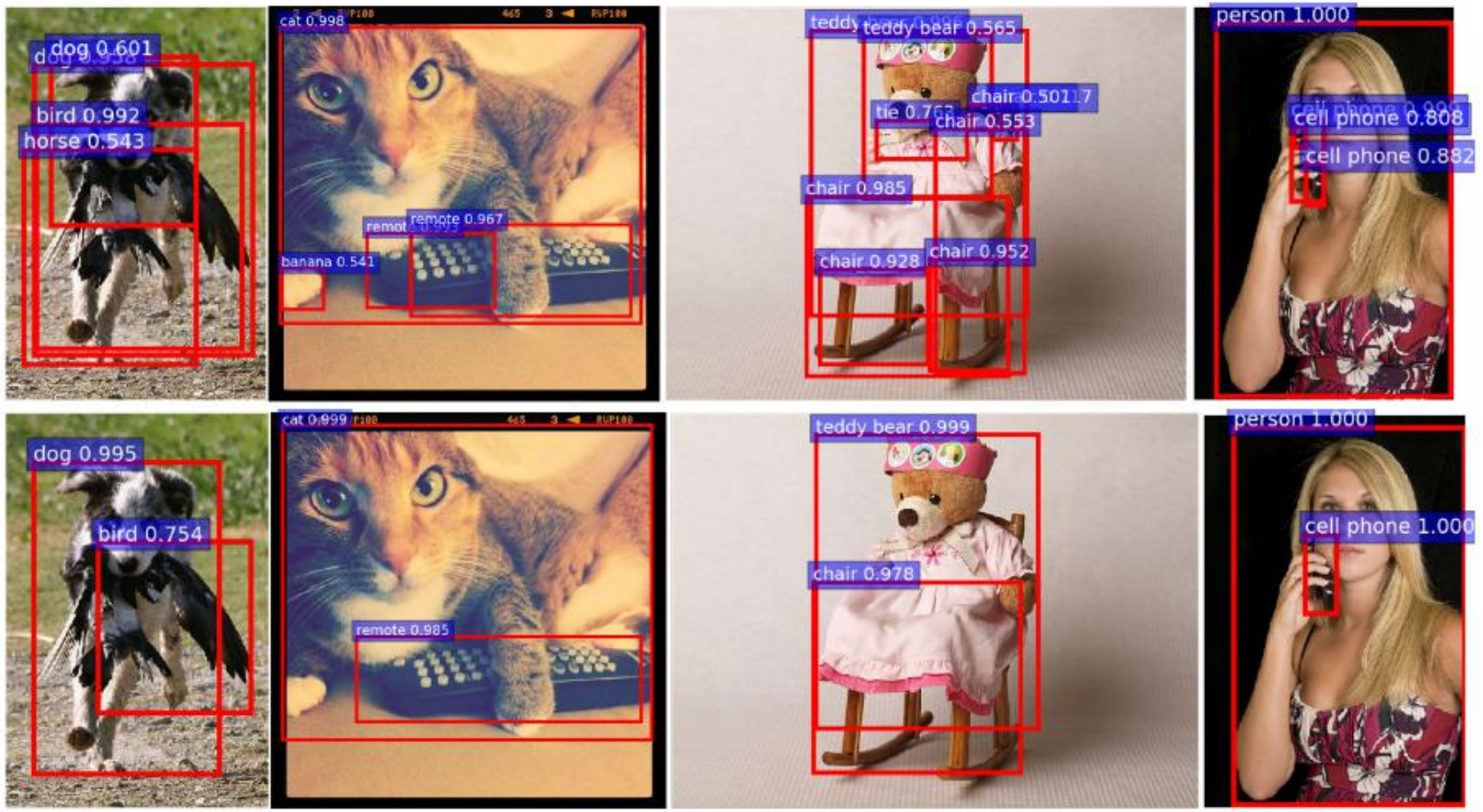


Figure 4: Comparisons of R-DAD without (top) /with (bottom) the region decomposition assembly method under occlusions on MSCOCO 2018 dataset.