

Multi-stage Object Detection with Group Recursive Learning

Jianan Li, Xiaodan Liang, Jianshu Li, Tingfa Xu, Jiashi Feng, and Shuicheng Yan, *Senior Member, IEEE*

Abstract—Most of existing detection pipelines treat object proposals independently and predict bounding box locations and classification scores over them separately. However, the important semantic and spatial layout correlations among proposals are often ignored, which are actually useful for more accurate object detection. In this work, we propose a new EM-like group recursive learning approach to iteratively refine object proposals by incorporating such context of surrounding proposals and provide an optimal spatial configuration of object detections. In addition, we propose to incorporate the weakly-supervised object segmentation cues and region-based object detection into a multi-stage architecture in order to fully exploit the learned segmentation features for better object detection in an end-to-end way. The proposed architecture consists of three cascaded networks which respectively learn to perform weakly-supervised object segmentation, object proposal generation and recursive detection refinement. Combining the group recursive learning and the multi-stage architecture provides competitive mAPs of 78.6% and 74.9% on the PASCAL VOC2007 and VOC2012 datasets respectively, which outperforms many well-established baselines [10] [20] significantly.

I. INTRODUCTION

Object detection is a fundamental problem in computer vision research. In recent years, remarkable progress has been made in object detection [16], [7], [26], [21], arguably benefited from the rapid development of deep neural network based methods [13], [23], [11], [10], [19], [4]. Among them, one of the most influential methods is the R-CNN framework [11] which performs CNN-based classification on the object proposals produced by various methods (*e.g.* [24] [27]). As two examples of improvement upon R-CNN, Fast R-CNN [10] learns a convolutional feature map from the entire image before extracting features to classify each proposal, and faster R-CNN [20] combines Region Proposal Network (RPN) and Fast R-CNN with shared convolutional layers. Those two variants both bring compelling accuracy and efficiency enhancement for object detection. However, existing R-CNN based methods make predictions for each proposal independently, although surrounding proposals of the same object can provide useful information to refine the proposal location to better cover the object. Moreover, they do not consider segmentation cues which are beneficial for better localizing the objects. In this paper, we aim to further enhance object detection by adopting two strategies, *i.e.* multi-stage network cascades and group recursive learning for detection refinement.

Jianan Li and Tingfa Xu are with School of Optical Engineering, Beijing Institute of Technology University, China. Xiaodan Liang is from Sun Yat-Sen University, China. Jianshu Li, Jiashi Feng and Shuicheng Yan are from Department of Electrical and Computer Engineering, National University of Singapore.

a) Multi-Stage Network Cascades: Object detection aims to tightly localize objects of particular categories in an image, while semantic segmentation aims to predict the category label for every pixel of the image. Although the two tasks are typically addressed separately, we argue that the features learned for semantic segmentation tasks could provide valuable cues for more accurately localizing objects — especially for the ones with small scale or occlusion. Therefore, we propose a multi-stage network cascades architecture to jointly perform weakly supervised semantic segmentation and object detection. The proposed architecture consists of three cascaded networks. The first network is for weakly supervised segmentation and learns specific semantic segmentation features from the entire image. The second network generates object proposals by considering both the convolutional features and the produced segmentation features. Better object proposals can thus be generated as foreground and background can be better distinguished using the segmentation related cues. Since there exist large variations in the initial locations of the produced proposals, it is usually hard to make precise predictions for some of the proposals independently with only one step. Thus the third network refines detections recursively based on object proposals produced in the previous stage and global dependency among multiple proposals. In this cascade way, the underlying segments from the semantic segmentation task which can provide local cues for better localization can be inherently integrated for object proposal generation and bounding box prediction. Moreover, precise predictions can be progressively obtained through recursive refinement using the global cues from multiple proposals.

b) Group Recursive Learning: Most existing approaches for object detection perform category prediction for each object proposal independently without considering the proposals in the vicinity. However, the mutual information among a group of neighboring proposals is quite valuable for getting more accurate detection results. As illustrated in Figure 1, although all of the object proposals have a large overlap with the ground truth, their relative locations to the ground truth and the semantic regions covered are significantly different. Some of the proposals are distant from the ground truth. It becomes difficult for the network to make precise predictions independently with such rough locations. One can observe that for a specific proposal, its surrounding proposals cover different parts of the object. They can provide useful cues to refine the proposal for better concentrating around the actual objects of interest.

Following the above intuition, we propose a group recursive

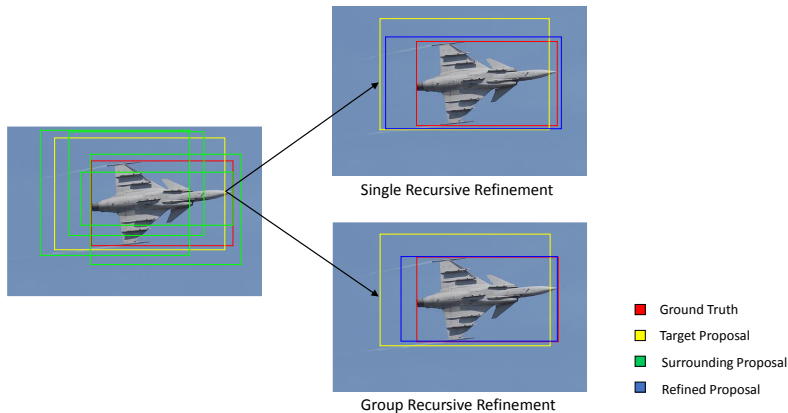


Fig. 1. Illustration of group recursive refinement. The red bounding box represents the ground truth location of an airplane. The yellow rectangle and the green rectangles denote the target object proposal to be refined and its surrounding object proposals of the same object. The blue rectangles represent the refined bounding box locations of the target object proposal. Compared to the refined location produced by regressing the target object proposal singly, a more accurate bounding box which tightly encloses the ground truth can be obtained from the group recursive refinement thanks to the useful location cues provided by multiple proposals.

learning approach to progressively refine object detection results in an expectation-maximization like manner. More concretely, in the E-step, given initial detection results, our proposed approach further refines each proposal by taking into consideration the surrounding proposals which have large overlap with the proposal of interest. These proposals are considered so that a *group* is formed. All the proposals within the same group collectively refine the proposal of interest to more precise locations. In the M-step, the likelihood of proposals being close to the corresponding ground truth bounding boxes is maximized through the learning process which provides more precise location predictions. This proposed recursive learning procedure is performed iteratively until optimal predictions are achieved.

c) Contributions: To summarize, we make the following contributions. (1) We develop a unified multi-stage network cascades architecture that is capable of leveraging semantic segmentation features for object detection. (2) We introduce an EM-like group recursive learning approach to iteratively refine detection results and minimize the offsets between object proposals and the ground truth step by step considering the global dependency among multiple proposals. (3) Our detection architecture achieves competitive mAPs of 78.6% and 74.9% on VOC2007 and VOC2012 detection challenges [6] respectively, which outperforms many well established baselines significantly.

II. RELATED WORK

In recent years, several works have proposed to incorporate segmentation techniques to assist object detection in different ways. For example, Parkhi *et al.* [18] improved the predicted bounding box with color models from predicted rectangles on cat and dog faces. Dai *et al.* [5] proposed to use segments extracted for each object detection hypothesis to accurately localize detected objects. Other research has exploited segmentation to generate object detection hypothesis for better localization. Segmentation was adopted as a selective search strategy to generate the best locations for object recognition in [25]. Arbelaez *et al.* [1] proposed a hierarchical segmenter

that leverages multiscale information and a grouping algorithm to produce accurate object candidates. Instead of using segmentation for better localizing detections, Fidler *et al.* [8] took advantage of semantic segmentation results [3] to more accurately score detections. In this work, we propose a unified framework to incorporate semantic segmentation features for both object proposal generation and better scoring and localizing detections. In addition, a group recursive learning strategy is employed to recursively refine the scores and locations of the detections, thus achieving more precise predictions.

III. OVERVIEW ON MULTI-STAGE OBJECT DETECTION ARCHITECTURE

Our proposed object detection architecture consists of a cascade of multiple CNN networks, each of which focuses on a specific task, *i.e.*, weakly-supervised semantic segmentation, proposal generation and recursive detection refinement respectively. The three networks share convolutional features learned from the entire image. Details about the proposed architecture are shown in Figure 2. The input image first passes through several convolutional and max pooling layers to produce convolutional feature maps. Then the semantic segmentation network learns semantic segmentation features for the entire image from the convolutional feature maps. The produced features are then fed into the proposal generation network to generate candidate object proposals. Finally, the recursive detection network iteratively refines the scores and locations of generated object proposals via a group recursive learning strategy. In the following subsections, we explain the multi-stage network cascades, group recursive learning scheme and testing phase with more details.

A. Multi-Stage Network Cascades

Object detection and semantic segmentation are two closely related tasks. The segments extracted for each object proposal can provide useful local cues (*e.g.*, object boundaries) for better object localization. In order to incorporate semantic segmentation cues to assist object detection, we introduce the

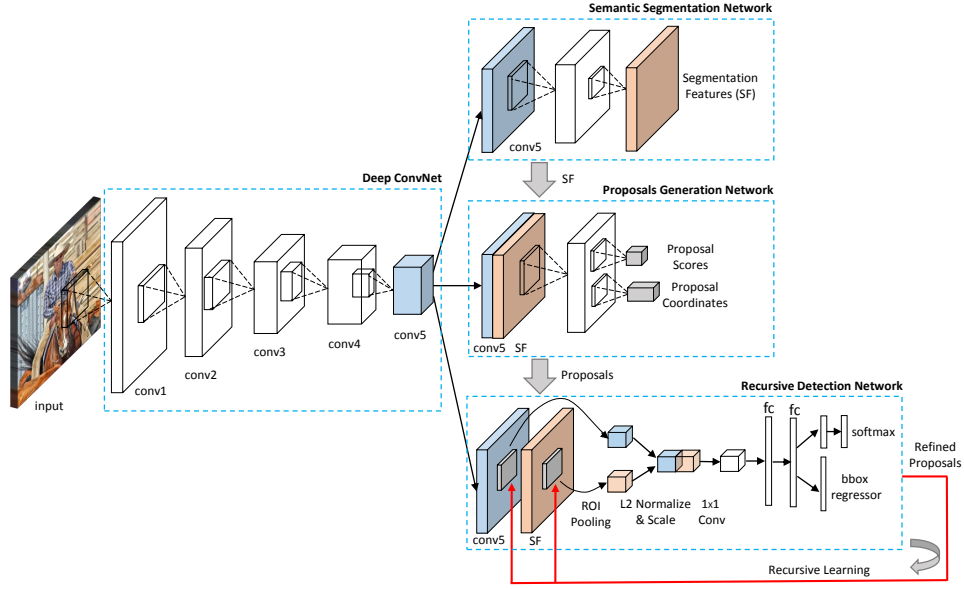


Fig. 2. Detailed architecture of the proposed framework. The whole input image is first fed into several convolutional layers and max pooling layers to generate the shared convolutional feature maps. The semantic segmentation network takes as input the shared feature maps and further computes the semantic segmentation feature maps for the input image through several convolutional layers. These computed feature maps are concatenated with the shared convolutional feature maps, forming the input of the proposal generation network to generate object proposals. For each produced proposal, the recursive detection network extracts a descriptor with fixed resolution from both types of feature maps using ROI pooling [10]. Each descriptor is L2-normalized, concatenated, scaled, and dimension-reduced (1×1 convolution) to produce a fixed-length feature descriptor of size $512 \times 7 \times 7$, which is fed into two fully connected layers to predict the confidences of all categories and the bounding box offsets. In addition, a group recursive learning scheme is performed to refine the bounding box locations and classification scores with multiple iterations. In each iteration, the bounding box locations for each proposal are refined by the predicted bounding box offsets and are further updated using the locations of its surrounding proposals of the same object through group refinement for more precise locations.

multi-stage network cascades architecture to jointly perform weakly-supervised semantic segmentation and object detection, in order to learn better image representations for object detection.

1) Weakly-Supervised Semantic Segmentation Network:

For the semantic segmentation network, we use the semantic segmentation-aware CNN model adopted in [9] which is trained for the class-specific foreground segmentation task based on a Fully Convolutional Network [17]. To avoid using additional segmentation annotations, the network is trained to predict class specific foreground probabilities in a weakly supervised manner with only the provided bounding box annotations for the detection task. The artificial foreground class specific segmentation masks are created using bounding boxes annotations. Specifically, the ground truth bounding boxes of an image are projected on the last hidden layer of the Fully Convolutional Network. The “pixels” inside the projected boxes are labeled as foreground while the rest are labeled as background. This process is performed independently for each class. After the Fully Convolutional Network has been trained on the class-specific foreground segmentation task, we drop the last classification layer and extract the convolutional feature maps output by the last convolutional layer as semantic segmentation features for the input images.

2) *Proposal Generation Network*: Based on the computed feature maps of the input image, the proposal generation network aims to produce a set of object proposals, each of which has a predicted objectness score. Following the Region Proposal Network (RPN) proposed in [20], the proposal generation network is structured with a convolutional layer fol-

lowed by a box-regression layer and a box-classification layer. Different from RPN [20], we incorporate the features learned from the semantic segmentation task which can provide better local cues for objectness prediction and proposal localization. Specifically, we concatenate the semantic segmentation feature maps produced by the semantic segmentation network and the last shared convolutional feature maps along the channel axis, forming the input of the proposal generation network. We minimize an objective function following the multi-task loss in [20] to optimize the parameters of the network.

3) *Recursive Detection Network*: The structure of the recursive detection network is based on the VGG-16 model [22], which aims to score the input object proposals and refine their bounding box locations following the Fast R-CNN detection pipeline [10]. Different from Fast R-CNN, segmentation-aware features are constructed to incorporate guidance from the pixel-wise segmentation information which can help better depict the boundaries of the objects to facilitate detection. Specifically, the recursive detection network first utilizes an ROI pooling layer to generate a fixed-length feature descriptor of size $7 \times 7 \times 512$ from both the semantic segmentation feature maps and the last shared convolutional feature maps for each proposal provided by the proposal generation network. Then, following the feature combination scheme adopted in [2], we concatenate each pooled feature descriptor along the channel axis and reduce the dimension with a 1×1 convolution to match the shape of $7 \times 7 \times 512$ required by the first fully-connected layer (fc6) of the pre-trained VGG-16 model. To match the original amplitudes, each pooled feature map is L2 normalized and re-scaled back up by a fixed scale of 1000.

The generated feature is then fed into two fully-connected layers (fc6 and fc7) to predict the confidences over $K + 1$ categories, including K object classes and one background class, as well as the bounding-box regression offsets. The parameters of these predictors are optimized by minimizing soft-max loss and smooth L1 loss [10].

B. Group Recursive Learning: An Expectation-Maximization Perspective

The group recursive learning works in an expectation-maximization like way, where the network parameter learning and group recursive refinement are alternatively performed. In particular, in the maximization step, the network is trained to minimize the loss function or equivalently maximize the likelihood of multiple object bounding box predictions. In the expectation step, the locations of the proposals are refined with induced group information. We now proceed to provide more details about the EM-like group recursive learning.

1) *The M-Step: Mini-Batch Gradient Descent:* Specifically, the initial object proposal is denoted as l where $l = (l_x, l_y, l_w, l_h)$ specifies its pixel coordinates of the center (l_x, l_y) and its width and height in pixels (l_h, l_w) . Each ground-truth bounding box l^* is specified in the same way: $l^* = (l_x^*, l_y^*, l_w^*, l_h^*)$. The bounding box regression targets r^* are computed as $r^* = f(l, l^*)$ following the transformation strategy $f(\cdot)$ adopted in [11], in which r^* specifies a scale-invariant translation and log-space height/width shift relative to an object proposal. In the t -th iteration, the network takes the refined bounding boxes l_{t-1} produced in the $(t - 1)$ -th iteration as input, and predicts bounding-box regression offsets, $r_{t,k} = (r_{t,k}^x, r_{t,k}^y, r_{t,k}^w, r_{t,k}^h)$ for each of the K object classes, indexed by k , and the category-level confidences $p_t = (p_{t,0}, \dots, p_{t,k})$ for $K + 1$ categories. Each training proposal is labeled with a ground-truth class g and a ground-truth bounding-box regression target r_t^* . We use a multi-task loss J on each object proposal to jointly train for classification and bounding-box regression:

$$J_t = J_{cls}(p_t, g) + \mathbf{1}[g \geq 1]J_{loc}(r_{t,g}, r_t^*), \quad (1)$$

where J_{cls} and J_{loc} are the losses for the classification and the bounding-box regression, respectively. In particular, $J_{cls}(p_t, g) = -\log p_{t,g}$ is log loss for the ground truth class g and J_{loc} is a smooth L_1 loss proposed in [10]. The Iverson bracket indicator function $\mathbf{1}[g \geq 1]$ equals 1 when $g \geq 1$ and 0 otherwise. For background proposals (*i.e.* $g = 0$), the J_{loc} is ignored. After the training process, the loss J in the t -th iteration will be minimized and the likelihood of the regressed proposals being near to the corresponding ground truth is maximized.

2) *The E-Step: Group Confidence Pooling:* The regressed bounding box l_t of the proposal can be computed as $f^{-1}(l_{t-1}, r_{t,g})$, where $f^{-1}(\cdot)$ represents the inverse operation of $f(\cdot)$. The final bounding box coordinates are further refined by considering the locations of all the surrounding proposals at different parts of the same object through a group confidence pooling scheme. Specifically, for a specific refined proposal $l_{t,i}$, denote D_t as the set of proposals of the same class that

have an overlap with $l_{t,i}$ of more than 0.7 on IOU metric. The refined location of $l_{t,i}$ can be taken as the expected location of the group by regarding the confidence score $s_{t,j}$ of each proposal $l_{t,j} \in D_t$ as a weight:

$$l'_{t,i} = \frac{\sum_{j:l_{t,j} \in D_t} s_{t,j} \cdot l_{t,j}}{\sum_{j:l_{t,j} \in D_t} s_{t,j}}. \quad (2)$$

With this group confidence pooling scheme, the proposals will be refined to a better location by taking the surrounding proposals into consideration. The better localized proposals will be given higher confidence scores. As a result, both loss terms in Eqn. (1) will be reduced.

Both the M-step and the E-step optimization can be realized within an end-to-end framework. Assume that the total number of refinement iterations is T . During the optimization, we unroll the detection network by stacking T detection networks with *shared* parameters. The global loss is computed as

$$J = \sum_{t \leq T} J_t + J_{pgn}, \quad (3)$$

where J_t (ref. Eqn. (1)) represents the loss produced by the recursive detection network at the t -th iteration with refined proposals and J_{pgn} denotes the loss output by the proposal generation network following the multi-task loss in [20]. Thus the multi-stage network cascades with group recursive learning can be trained end-to-end jointly.

C. Testing

In testing, given an input image, the proposed framework first generates initial object proposals using the proposal generation network and then recursively passes them into the recursive detection network. At the t -th iteration, the recursive detection network predicts the category-level confidences p_t and bounding-box regression offsets r_t for each proposal. The category of the proposal is predicted as the class with the maximum score in p_t . For the proposals predicted as a specific object class, the locations of the proposals are updated by refining the previous location l_{t-1} with the predicted bounding-box regression offsets $r_{t,g}$ and then performing the group confidence pooling scheme as previously mentioned. For the proposals predicted as the background class, the locations of the proposals are not updated. The final outputs for each proposal are the results in the last iteration $t = T$, including the predicted category-level confidences p_T and the refined locations l_T .

IV. EXPERIMENTS

A. Experimental Settings

a) *Datasets and Evaluation Metrics:* To make fair comparison with the state-of-the-art methods [10] [20] [9], we evaluate the proposed framework on the PASCAL VOC 2007 benchmark and PASCAL VOC 2012 benchmark [6]. The two datasets consist of 9,963 and 22,531 images respectively, and they are divided into train, val and test subsets. The model evaluated on VOC 2007 is trained based on the trainval split from VOC 2007, including 5,011 images, and the trainval

TABLE I

DETECTION RESULTS ON VOC 2007 TEST. **P**: INCORPORATE SEMANTIC FEATURES FOR OBJECT PROPOSAL GENERATION, **D**: INCORPORATE SEMANTIC FEATURES FOR OBJECT CLASSIFICATION AND BOUNDING BOX REGRESSION, **R**: PERFORM GROUP RECURSIVE LEARNING.

Method	P D R	mAP	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
FRCN [10]		70.0	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
RPN [20]		73.2	76.5	79.0	70.9	65.5	52.1	83.1	84.7	86.4	52.0	81.9	65.7	84.8	84.6	77.5	76.7	38.8	73.6	73.9	83.0	72.6
ResNet-101 [12]		76.4	79.8	80.7	76.2	68.3	55.9	85.1	85.3	89.8	56.7	87.8	69.4	88.3	88.9	80.9	78.4	41.7	78.6	79.8	85.3	72.0
MR-CNN [9]		78.2	80.3	84.1	78.5	70.8	68.5	88.0	85.9	87.8	60.3	85.2	73.7	87.2	86.5	85.0	76.4	48.5	76.3	75.5	85.0	81.0
Ours (Baseline)		76.0	78.6	80.1	77.7	67.0	63.2	86.1	87.9	89.0	58.7	82.4	70.6	84.7	87.1	76.9	79.0	47.2	75.4	70.6	82.5	74.7
Ours	✓	76.4	79.4	79.9	76.5	69.3	62.8	86.8	87.5	88.5	58.2	83.3	71.4	84.7	85.2	78.9	78.8	49.1	77.2	70.5	83.8	75.4
Ours	✓✓	77.6	78.7	85.7	76.8	71.8	64.7	85.7	87.5	87.7	60.2	85.2	72.5	87.0	86.7	79.6	79.3	48.8	76.6	77.2	84.3	75.9
Ours	✓✓✓	78.6	80.0	81.0	77.4	72.1	64.3	88.2	88.1	88.4	64.4	85.4	73.1	87.3	87.4	85.1	79.6	50.1	78.4	79.5	86.9	75.5

TABLE II

DETECTION RESULTS ON VOC 2012 TEST. **P**: INCORPORATE SEMANTIC FEATURES FOR OBJECT PROPOSAL GENERATION, **D**: INCORPORATE SEMANTIC FEATURES FOR OBJECT CLASSIFICATION AND BOUNDING BOX REGRESSION, **R**: PERFORM GROUP RECURSIVE LEARNING.

Method	P D R	mAP	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
FRCN [10]		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
RPN [20]		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
FRCN+YOLO [19]		70.4	83.0	78.5	73.7	55.8	43.1	78.3	73.0	89.2	49.1	74.3	56.6	87.2	80.5	80.5	74.7	42.1	70.8	68.3	81.5	67.0
HyperNet		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
ResNet-101 [12]		73.8	86.5	81.6	77.2	58.0	51.0	78.6	76.6	93.2	48.6	80.4	59.0	92.1	85.3	84.8	80.7	48.1	77.3	66.5	84.7	65.6
MR-CNN [9]		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
Ours	✓✓✓	74.9	85.7	82.0	75.0	62.7	58.3	80.5	80.3	89.4	55.8	78.2	62.7	87.2	83.2	84.3	82.7	53.4	76.0	67.5	83.7	70.4

split from VOC 2012, including 11,540 images. The model evaluated on VOC 2012 is trained based on all images from VOC 2007, including 9,963 images, and the trainval split from VOC 2012. We use standard evaluation metrics Average Precision (AP) and mean of AP (mAP) following the PASCAL challenge protocols for evaluation.

b) Implementation Details: We initialize the bottom shared convolutional layers and the recursive detection network with the pre-trained VGG-16 model [22] and initialize the semantic segmentation network with the pre-trained semantic segmentation-aware CNN model in [9]. All the other newly added layers are initialized by drawing weights from a zero-mean Gaussian distribution with standard deviation 0.01 and 0.001. Our code is based on the publicly available Faster R-CNN framework [20] built on the Caffe platform [15]. We fine-tune the whole framework jointly following the fine-tuning strategy proposed in [20]. During fine-tuning, images are randomly selected for horizontally flipping with a probability of 0.5 to augment the training data. We set the iteration number for group recursive learning as $T = 2$, since only minor improvement with more iterations is observed. We run Stochastic Gradient Descent (SGD) for totally 140k iterations to train the network parameters for VOC 2007 and VOC 2012. The initial learning rate of all layers is set as 0.001 and decreased to one tenth of the current rate after 80k iterations. The model is trained on a NVIDIA GeForce Titan X GPU and Intel Core i7-4930K CPU @ 3.40 GHz.

B. Performance Comparisons

Table I and Table II provide the comparisons of the proposed framework with several state-of-the-art methods [10] [20] [9] [19]. It can be observed that our method obtains the mAP score of 78.6% on VOC 2007, which outperforms the two baselines by 8.6% for Girshick *et al.* [11]

and 5.4% for Ren *et al.* [20]. On VOC 2012, our method outperforms the two baselines: 74.9% vs 68.4% of Girshick *et al.* [11] and 70.4% of Ren *et al.* [20]. In general, the proposed method shows significantly higher performance compared with the baselines and achieves competitive results compared with the state-of-the-art methods on both datasets, which validates its superiority in accurate object detection benefited from the multi-stage network cascades framework and the group recursive learning strategy.

C. Ablation Studies

We further evaluate two important components, *i.e.* multi-stage network cascades and group recursive learning, to validate their effectiveness.

c) Multi-stage Network Cascades: We verify the effectiveness of incorporating semantic segmentation features for better object proposal generation and detection using the multi-stage network cascades framework. As shown in Table I, 0.4% improvement can be observed by incorporating the semantic segmentation features into the proposal generation network compared to the variant without using semantic segmentation features where object proposals and detection results are directly generated based on the last shared convolutional features. Similarly, incorporating the semantic segmentation features into the object detection network offers a further performance increase of 1.2%. This demonstrates that the proposed multi-stage network cascades framework can effectively leverage the learned features from the semantic segmentation task for object detection, which leads to more accurate bounding boxes for object proposals and provides useful local cues for better object classification and localization.

d) Group Recursive Learning: In the proposed method, we set the maximal number of iterations for group recursive learning as $T = 2$. To verify the effectiveness of the proposed

TABLE III
COMPARISON OF PERFORMANCE WITH SEVERAL ARCHITECTURAL VARIANTS OF OUR PROPOSED FRAMEWORK ON VOC 2007 TEST.

Method	mAP	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
Iter_1	77.6	78.7	85.7	76.8	71.8	64.7	85.7	87.5	87.7	60.2	85.2	72.5	87.0	86.7	79.6	79.3	48.8	76.6	77.2	84.3	75.9
Iter_2	78.6	80.0	81.0	77.4	72.1	64.3	88.2	88.1	88.4	64.4	85.4	73.1	87.3	87.4	85.1	79.6	50.1	78.4	79.5	86.9	75.5
Iter_2_testing	77.8	80.0	80.8	76.7	72.4	63.9	85.4	88.0	89.1	59.5	85.1	74.5	86.7	86.7	79.9	79.5	49.9	77.6	78.1	85.2	76.0

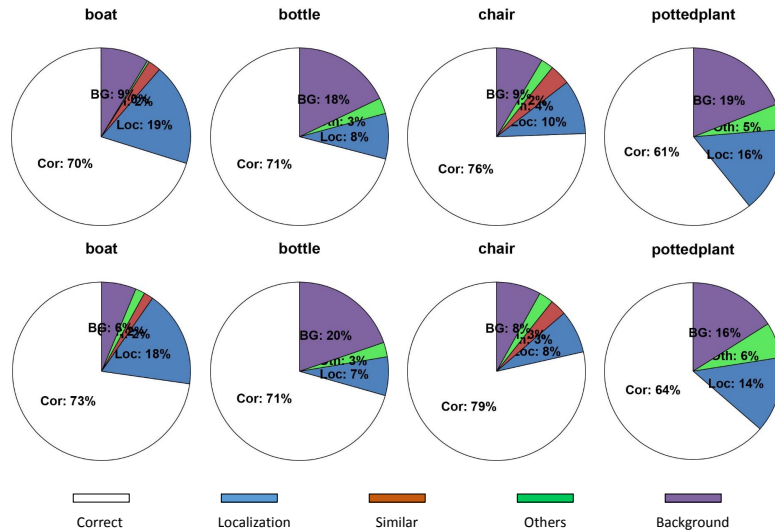


Fig. 3. Analysis of top ranked false positives on VOC 2007 test. Fractions of top N detections (N is the number of objects in the category) that are correct (Cor), or false positives due to poor localization (Loc), confusion with similar objects (Sim), confusion with other VOC objects (Oth), or confusion with background or unlabeled objects (BG), are shown. We only show the graphs for challenging classes, *i.e.* *boat*, *bottle*, *chair* and *pottedplant*, due to space limitations. **Top row**: the results of our baseline model. **Bottom row**: the results of the proposed method.

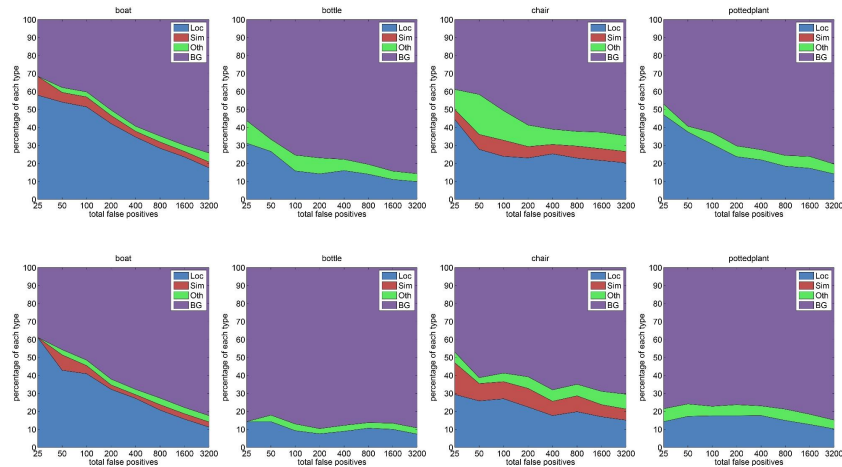


Fig. 4. Top ranked false positive types on VOC 2007 test. We only show the graphs for challenging classes, *i.e.* *boat*, *bottle*, *chair* and *pottedplant*, due to space limitations. **Top row**: the results of the baseline model. **Bottom row**: the results of the proposed method.

group recursive learning scheme, we evaluate the performance of the proposed framework with different numbers of iterations during the training and testing stage. In Tabel III, “Iter_1” denotes the variant without using any recursive refinement where detection results are generated with only 1 iteration and Iter_2” represents the model of using 2 iterations. Compared with “Iter_1”, Iter_2 improves the performance by 1.0%, which verifies that more precise detection results can be obtained benefited from the recursively refined bounding box locations and classification scores. Since no noticeable

improvement can be observed by adding more iterations, we use 2 iterations for group recursive learning throughout our experiments.

To verify the advantage of using group recursive learning scheme in both the training and testing stage, we evaluate the performance of the variant where the recursive process is only performed during the testing stage, denoted as “Iter_2_testing”. As shown in Tabel III, a 0.8% drop in performance is observed by comparing “Iter_2_testing” with

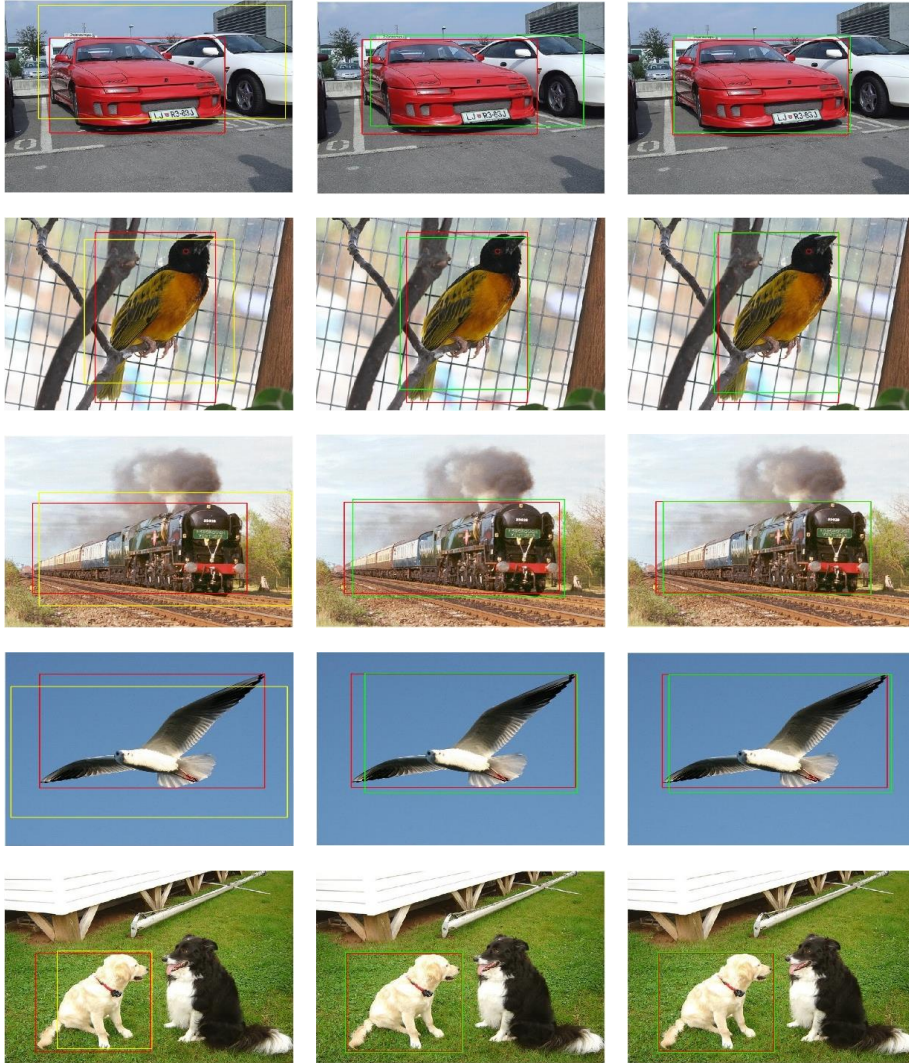


Fig. 5. Qualitative results of the iterative bounding box location refinement procedure given an initial object proposal. The ground-truth bounding boxes of objects are annotated with red rectangles. The yellow and green rectangles represent the initial object proposal produced by the proposal generation network and the refined bounding box location from each refinement iteration, respectively.

”Iter_2”, demonstrating that employing group recursive refinement during both the training and testing stage is beneficial for jointly improving the network capabilities.

D. Detection Error Analysis

We analyze the detection errors of the proposed method using the tool of Hoiem *et al.* [14]. In Figure 3, we plot pie charts with the percentage of detections that are false positives due to bad localization, confusion with similar categories and other categories, and confusion with background or unlabeled objects. It can be observed that the proposed framework achieves a considerable reduction in the percentage of false positives due to bad localization for challenging categories. This validates that incorporating semantic segmentation features can increase the localization sensitivity of the detection network and precise bounding boxes for the detections can be obtained by adopting the proposed group recursive learning scheme. The similar observation can be deduced from Fig-

ure 4 where we plot the top-ranked false positive types of the baseline and of the proposed framework.

E. Qualitative Results

In Figure 5, we provide sample qualitative results that present the iterative bounding box location refinement procedure starting from an initial object proposal produced by the proposal generation network. This example shows that our proposed method is capable of refining the produced initial object proposals step by step to fit them to the ground-truth bounding boxes of different objects, providing accurate object localization.

V. CONCLUSION

In this paper, we propose a multi-stage network cascades framework with group recursive learning for object detection. Specially, the proposed framework effectively utilizes semantic segmentation features to assist object detection by incorporating the semantic segmentation network, proposal generation

network and recursive detection network into a unified architecture. In addition, a group recursive learning scheme is proposed to recursively score object proposals and regress their bounding boxes considering the locations of the surrounding proposals of the same object. We show that the proposed framework is particularly effective in object localization and achieves competitive results on PASCAL VOC 2007 and 2012.

REFERENCES

- [1] Pablo Arbelaez, Jordi Pont-Tuset, Jon Barron, Ferran Marques, and Jitendra Malik. Multiscale combinatorial grouping. In *CVPR*, pages 328–335, 2014.
- [2] Sean Bell, C Lawrence Zitnick, Kavita Bala, and Ross Girshick. Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks. *arXiv preprint arXiv:1512.04143*, 2015.
- [3] Joao Carreira, Rui Caseiro, Jorge Batista, and Cristian Sminchisescu. Semantic segmentation with second-order pooling. In *ECCV*, pages 430–443, 2012.
- [4] Xiaozhi Chen, Kaustav Kundu, Yukun Zhu, Andrew G Berneshawi, Huimin Ma, Sanja Fidler, and Raquel Urtasun. 3d object proposals for accurate object class detection. In *NIPS*, pages 424–432, 2015.
- [5] Qieyun Dai and Derek Hoiem. Learning to localize detected objects. In *CVPR*, pages 3322–3329, 2012.
- [6] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010.
- [7] Sanja Fidler, Sven Dickinson, and Raquel Urtasun. 3d object detection and viewpoint estimation with a deformable 3d cuboid model. In *NIPS*, pages 611–619, 2012.
- [8] Sanja Fidler, Roozbeh Mottaghi, Alan Yuille, and Raquel Urtasun. Bottom-up segmentation for top-down detection. In *CVPR*, pages 3294–3301, 2013.
- [9] Spyros Gidaris and Nikos Komodakis. Object detection via a multi-region & semantic segmentation-aware cnn model. In *ICCV*, 2015.
- [10] Ross Girshick. Fast r-cnn. In *CVPR*, pages 1440–1448, 2015.
- [11] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jagannath Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, pages 580–587, 2014.
- [12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.
- [13] Judy Hoffman, Sergio Guadarrama, Eric S Tzeng, Ronghang Hu, Jeff Donahue, Ross Girshick, Trevor Darrell, and Kate Saenko. Lsda: Large scale detection through adaptation. In *NIPS*, pages 3536–3544, 2014.
- [14] Derek Hoiem, Yodsawalai Chodpathumwan, and Qieyun Dai. Diagnosing error in object detectors. In *ECCV*, pages 340–353, 2012.
- [15] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM Multimedia*, pages 675–678, 2014.
- [16] Peter Kontschieder, Samuel R. Bulò, Antonio Criminisi, Pushmeet Kohli, Marcello Pelillo, and Horst Bischof. Context-sensitive decision forests for object detection. In *NIPS*, pages 431–439, 2012.
- [17] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, pages 3431–3440, 2015.
- [18] Omkar M Parkhi, Andrea Vedaldi, CV Jawahar, and Andrew Zisserman. The truth about cats and dogs. In *ICCV*, pages 1427–1434, 2011.
- [19] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. *arXiv preprint arXiv:1506.02640*, 2015.
- [20] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, pages 91–99, 2015.
- [21] Mohammad Saberian and Nuno Vasconcelos. Multi-resolution cascades for multiclass object detection. In *NIPS*, pages 2186–2194, 2014.
- [22] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [23] Christian Szegedy, Alexander Toshev, and Dumitru Erhan. Deep neural networks for object detection. In *NIPS*, pages 2553–2561, 2013.
- [24] Jasper RR Uijlings, Koen EA van de Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *IJCV*, 104(2):154–171, 2013.
- [25] Koen EA Van de Sande, Jasper RR Uijlings, Theo Gevers, and Arnold WM Smeulders. Segmentation as selective search for object recognition. In *ICCV*, pages 1879–1886, 2011.
- [26] Xiaolong Wang and Liang Lin. Dynamical and-or graph learning for object shape modeling and detection. In *NIPS*, pages 242–250, 2012.
- [27] C Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, pages 391–405, 2014.