CAD: Scale Invariant Framework for Real-Time Object Detection

Huajun Zhou Zechao Li* Chengcheng Ning Jinhui Tang
School of Computer Science and Engineering
Nanjing University of Science and Technology
zechao.li@njust.edu.cn

Abstract

Real-time detection frameworks that typically utilize end-to-end networks to scan the entire vision range, have shown potential effectiveness in object detection. However, compared to more accurate but time-consuming frameworks, detection accuracy of existing real-time networks are still left far behind. Towards this end, this work proposes a novel CAD framework to improve detection accuracy while preserving the real-time speed. Moreover, to enhance the generalization ability of the proposed framework, we introduce maxout [1] to approximate the correlation between image pixels and network predictions. In addition, the non-maximum weighted (NMW) [2] is employed to eliminate the redundant bounding boxes that are considered as repetitive detections for the same objects. Extensive experiments are conducted on two detection benchmarks to demonstrate that the proposed framework achieves state-of-the-art performance.

1. Introduction

With the development of robotic applications, object detection, which is intended to locate all target objects in the visual range, has become a significant computer vision task with increasing attraction. The performance of object detection on ImageNet and PASCAL VOC has been significantly improved with the success of deep Convolutional Neural Networks (CNNs) [3, 4, 5, 6] in recent years. The pivotal challenge of object detection is to locate object regions within the images and distinguish object categories simultaneously.

To further increase the performance, several frameworks, such as region CNN (R-CNN) [7], have been proposed to tune deep neural networks for detection. R-CNN learns features for Regions-of-Interest (RoIs) by employing the deep neural network, and then classifies these learned features by multiple two-class SVMs. On the other hand, instead of warping various RoIs to original image size, SPP-Net [8] reshapes these RoIs to fixed-shape cubes by introducing the spatial pyramid pooling (SPP) layer. In Faster-RCNN [9], the region proposal network (RPN) outputs preliminary RoIs instead of traditional selective search method. Subsequently, a prevalent family of networks [10, 11, 12, 13, 14] have been proposed to increase speed and accuracy. However, essential multi-step methodologies of aforementioned frameworks limit the ceiling of their detection speed.

Recently, we have witnessed several end-to-end frame-

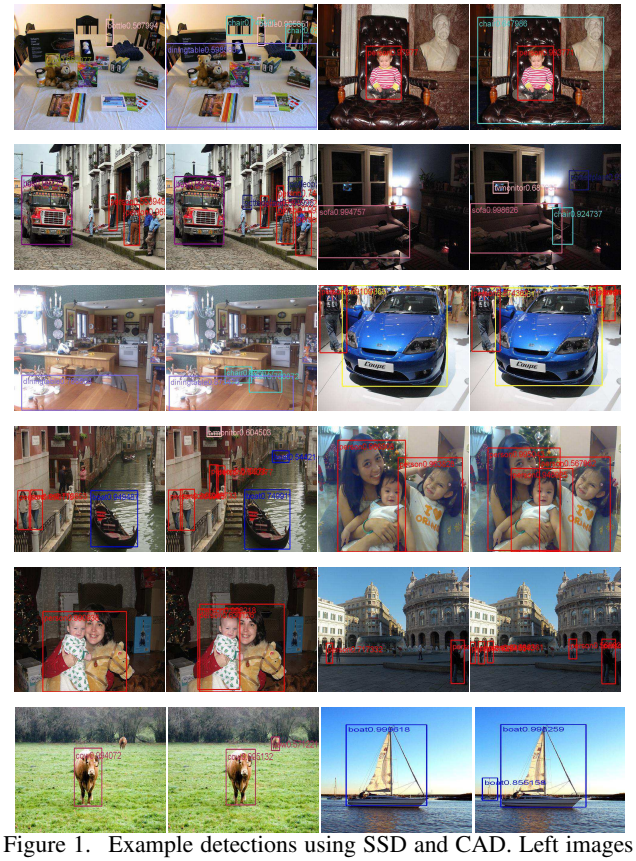

Figure 1. Example detections using SSD and CAD. Left images are detected by SSD while detections of our CAD framework are shown on the right.

*Corresponding author.

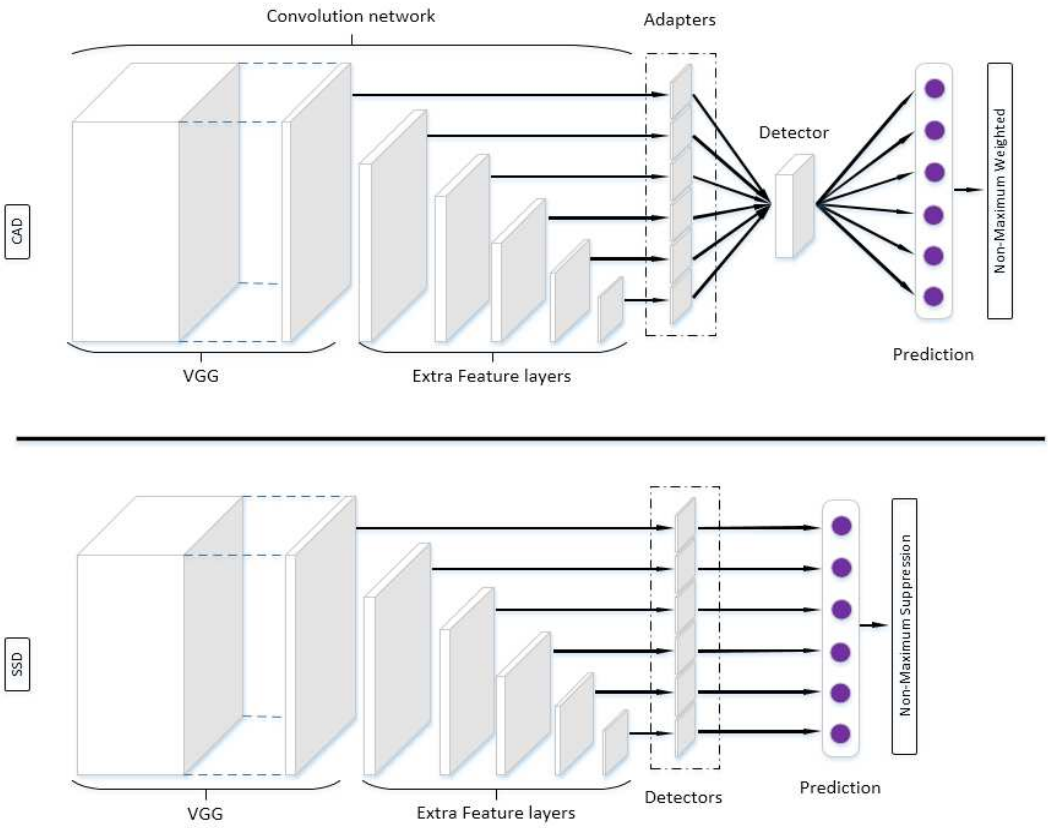

Figure 2. Framework architecture of CAD framework. We add Adapters and Detector to the end of a convolution network. In addition, the non-maximum weighted method eliminates redundant detections.

works [15, 16] achieve remarkable detection performance with surprising speed. YOLO [15] first detects objects using a single convolution network and achieves remarkable accuracy with impressive speed. To match various objects within images, SSD [16] generates extensive prior boxes base on the receptive fields of convolution layers. Moreover, YOLO9000 [17] and DSSD [18] further improve detection accuracy but decelerate their detection speed. Without consideration of the instinctive speed advantage, all end-to-end frameworks are perplexed to small objects and unfamiliar categories. Thus, compared to aforementioned time-consuming frameworks, overall accuracy of real-time detection frameworks still have considerable potential for further improvement.

Towards this end, we propose a novel end-to-end framework that improves detection performance while preserving real-time speed. Above all, three components (Convolution network, Adapters and Detector) are integrated into a unified network to accelerate detection speed. Convolution network generates prior boxes and associated feature vectors while Adapters recombine these vectors to harmonize with subsequent Detector. As the crucial component in our framework, Detector predicts category confidences and bounding box offsets base on the recombined vectors from Adapters. To strengthen the generalization ability of detection framework, we adopt maxout [1] in Detector to approximate the correlation between image pixels and network predictions. Finally, redundant prediction boxes are eliminated by weighted-averaging the bounding boxes that are seen as detections for the same objects. Using VGG [5] as backbone network, our framework with 300×300 input size yields 79.3% mAP on VOC2007 test and 76.9% mAP on VOC2012 test with 35FPS using our NVIDIA GTX1070.

We summarize the main contributions as follows:

- We develop a novel end-to-end framework that is significantly more accurate than existing end-to-end frameworks while preserving real-time speed.

- To better imitate the correlation between image pixels and framework predictions, maxout is employed to approximate more complex curves in the proposed framework.

- Redundant prediction boxes are post-processed by the non-maximum weighted method instead of conventional non-maximum suppression.

2. Related Work

In this section, we review several related works about RPN-based frameworks and end-to-end frameworks via deep convolution network.

2.1. RPN-based Frameworks

In recent years, we have witnessed incessant progress in object detection since the ground-breaking work of region convolution neural network (R-CNN) [7], which introduced convolution neuron networks (CNNs) into object detection task and defeated the traditional Deformable Parts Model (DPM) [19] method. On one hand, these advances owe to the progress in deep learning techniques [20, 21, 22]. On the other hand, detection performance benefits from the developments of detection methodology. Original R-CNN presented a three-step methodology to handle the detection task: region proposal, feature extraction [23], and regions classification. Firstly, a fixed number of Regions-of-Interest (RoIs) that are supposed to contain target objects are generated by selective search method [24]. Subsequently, all RoIs are clipped from the original images and delivered into a convolution network as individual images. Finally, multiple two-class SVMs produce the category confidences for all RoIs by distinguishing related feature vectors, which are generated by the above convolution network. Multibox [25] first applies bounding box regression for the detection task and, due to its remarkable improvement, inherited by subsequent networks. SPP-Net [8] substantially accelerates R-CNN by proposing the spatial pyramid pooling (SPP) layer that pools arbitrary regions to fixed-size cubes.

Based on aforementioned frameworks, follow-up Fast-RCNN [26] presented a new framework that fine-tunes all layers end-to-end by optimizing the objective function for both confidences and bounding box regression. Besides, Fast-RCNN developed a novel RoI pooling layer by simplifying the SPP layer. Moreover, a novel two-step methodology was proposed by integrating the last two steps of the three-step methodology in R-CNN into a unified convolution network. Faster-RCNN [9], the first end-to-end training framework with nearly real-time detection speed, achieves superior detection performance and profound influence to the whole object detection region. Instead of conventional selective search method, Faster-RCNN proposed the region proposal network (RPN) to generate preliminary RoIs, which becomes the most widespread region proposal method in current detection frameworks. RPN scans over the convolution feature maps by using a small network to produce anchors and related feature vectors that are used to predict category scores and coordinates for related anchors at each position. In addition, Faster-RCNN integrates the RPN with Fast-RCNN by alternating between fine-tuning shared convolutional layers and prediction layers for these two networks. Recent R-FCN network [27] achieves remarkable performance on object detection benchmarks by introducing vote strategy to Faster-RCNN.

2.2. End-to-end Frameworks

Compared to aforementioned mature RPN-based networks, end-to-end detection frameworks are in the ascendant. Overfeat [28] detects target objects by sliding multi-scale windows on the convolutional feature maps. YOLO [15] achieves astonishing detection speed while preserving appreciable detection accuracy. In YOLO, input images are divided into several square grids and delivered to a convolution network. Each grid is required to detect the objects that their center points fall into its grid region. Convolution network outputs a cube that indicates the predictions of these grids. Successive work YOLO9000 [17] improves YOLO by employing dimension clusters, multi-scale training and so on techniques to achieve superior performance on object detection benchmarks. SSD [16] constructs a multi-resolution pyramid by appending additional convolution layers with progressively smaller scales to a basic convolution network. And then, numerous prior boxes with different aspect ratios and areas are generated by this network to match the ground true boxes. Other auxiliary layers are appended to top layers of the pyramid to predict category confidences and bounding box offsets for all prior boxes. Subsequently, concatenated multi-scale predictions are post-processed by non-maximum suppression (NMS) to generate the final detection results. DSSD [18] further constructs the inverted feature map pyramid by adding some deconvolution layers to the topmost layers of the SSD network. Consequently, this strategy improves 1-2% mAP whereas decelerates the network approximately 20%.

3. CAD Framework

3.1. Region Generation

We argue that the object detection task can be explained from another perspective. For example, object location regression can be considered as a dynamic function $f(\boldsymbol{b})$ that indicates the maximum jaccard overlap between bounding box $\boldsymbol{b}$ and all ground true boxes:

$$f(\boldsymbol{b}) = \max iou(\boldsymbol{b}, \boldsymbol{b}_{gt})$$

Here $\boldsymbol{b}$ is a bounding box denoted as the 4-dim vector (x, y, w, h) that represents the center point coordinates, width and height. $\boldsymbol{b}_{gt}$ indicates the ground true boxes of original images while the iou function calculates the jaccard overlap of two input boxes. For any image containing target objects, there is a corresponding function $f^*(\boldsymbol{b})$ that exists some inputs $\boldsymbol{b}_{gt}$ such that:

$$f^*(\boldsymbol{b}_{gt}) = \max f^*(\boldsymbol{b}) = 1$$

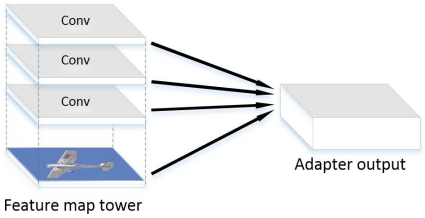

Figure 3. Structure of the feature map tower in experiments.

To imitate $f^*(\boldsymbol{b})$ that indicates the correlation between image pixels and function outputs, detection frameworks are trained using the known correlation instances - labeled images. In detection, based on the learned correlations, our objective is to approximate all the maximizers $\boldsymbol{b}_{gt}$ in the domain of definition.

To tackle this mathematical problem, a straightforward strategy that densely sample inputs from the domain of definition is employed to find out the sample points that are close to any maximizer. Afterwards, these chosen sample points are gathered and post-processed by specific methods to approximate all the maximizers. In detection, to match the ground true boxes, the proposed framework generates extensive prior boxes for dense search in the domain of definition. Additionally, the criterion of "match" and "close to" depends on whether the jaccard overlap of two boxes surpasses a given threshold.

In implementation, to adapt the detection task, we truncate the topmost full connected layers and pooling layers of a convolution network, which is trained on ImageNet for image classification. To construct the base network of the proposed framework, additional convolution layers with progressively decreasing sizes are appended to the top of the truncated convolution network. Subsequently, numerous prior boxes and related feature vectors are generated from the top layers of the base network according to their receptive fields. It is noteworthy that, in each layer, multiple prior boxes at the same location have different aspect ratios and areas but a common feature vector.

3.2. Feature Recombination

Since convolution networks typically have more convolutional filters in the intermediate layers, the lengths of feature vectors from different convolution layers are various in the base network. To cooperate with subsequent scale invariant detection, these various features have to be recombined to a unified length. Besides, in the base network, shallower and wider convolution layers are supposed to produce similar predictions as deeper and narrower layers. With more convolution layers below, deeper layers reorganize the original image pixels to higher-level representations, which are significant in both classification and regression. The proposed framework desires Adapters to compensate shallower layers and recombine the bottom feature vectors to unified length.

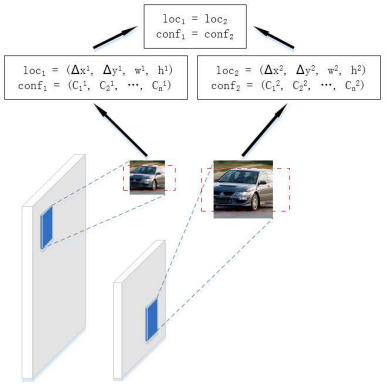

Figure 4. Illustration of scale invariant detection. If one object with different scales shown at different locations, Detectors for different scales are supposed to produce same predictions.

Therefore, we append feature map towers, the instantiation of our Adapters, to all chosen layers in the base network. Internal structure of these towers refer to Figure 3. All towers are compelled to output same shapes cube by restraining all layers to employ equivalent convolutional filters.

3.3. Scale Invariant Detection

Scale invariant detection (SID) indicates that Detectors should produce the same predictions from different resolution image patches. Illustration of SID please refer to Figure 4.

In SSD framework, multiple Detectors are trained to distinguish whether the maximum jaccard overlaps between different size patches and the ground true boxes are greater than the given threshold. Since boxes generally have higher overlaps with similar size boxes, larger objects are more likely to become positive examples in the Detectors that contain larger prior boxes. Consequently, all ground true boxes in training set are roughly divided into several groups base on their sizes. Compared to other computer vision datasets, object detection instances are quite scarce. However, this strategy further splits the dataset and allocates these pieces to several Detectors respectively. Detectors obtain far less training instances than existing dataset, which may induce them to over-fitting.

In addition, it is noteworthy that offset predictions are proportional to their prior box sizes as following formulas:

$$\begin{cases} \Delta x_{gt} = & \hat{x}_p \times w_{pbox} \\ \Delta y_{gt} = & \hat{y}_p \times h_{pbox} \\ w_{gt} = & e^{\hat{w}_p} \times w_{pbox} \\ h_{gt} = & e^{\hat{h}_p} \times h_{pbox} \end{cases}$$

Bounding boxes are encoded as (x, y, w, h), represents center coordinates (x, y), width and height. Δx_{gt} and Δy_{gt}

are the differences between ground true boxes and prior boxes. $(\hat{x}_p, \hat{y}_p, \hat{w}_p, \hat{h}_p)$ are expected predictions while (w_{pbox}, h_{pbox}) are width and height of the prior boxes. Although larger objects have larger bounding box offsets, these offsets are proportional to the prior box sizes. After divided by bounding box sizes, predictions for arbitrary size objects are equivalent. From the other perspective, as all Detectors are 3x3 convolution layers in practice, SID have the structural convenience for implementation. However, multiple Detectors with respective parameters indicate that SID is notoriously difficult to achieve.

To address these issues, a novel Detector implementation is designed for SID in the proposed framework. Firstly, just like SSD, Detectors are employed to predict C+1 (classes and background) category confidences and 4 bounding box offsets for corresponding prior boxes based on feature vectors from Adapters. Furthermore, all Detectors are merged to a single Detector by sharing their parameters, which ensure that they will produce identical predictions for different resolution image patches. If a large image patch is considered as positive example in large prior boxes Detector, the rest of Detectors will be trained using this instance too. In the sense of generalization, it is similar with data augmentation that increases positive examples for all Detectors. Moreover, with the consideration that the maps from feature space to bounding box offset space are supposed to be nonlinear, we apply maxout to enhance approximation ability of the Detector.

3.4. Non-Maximum Weighted

In existing detection frameworks, non-maximum suppression (NMS) is the most widespread method to eliminate the redundant prediction boxes. For all predictions, if the jaccard overlap between two prediction boxes surpasses a given threshold, they are identified as the detections for the same object and the higher confident one becomes the final prediction. Suppose $\boldsymbol{B}$ is a group of boxes that are identified as the same object detections and $\boldsymbol{b}_{pre}$ denotes the final prediction bounding box. NMS implements the following function:

$$\boldsymbol{b}_{pre} = \boldsymbol{b}_{\mathrm{argmax}_i \, \boldsymbol{c}_i}$$

Here $\boldsymbol{c}_i$ indicates the confidence of bounding box $\boldsymbol{b}_i$ in $\boldsymbol{B}$. For detection boxes of same object, this method simply adopts the most confident boxes while ignores all the non-maximum boxes.

Intuitively, lower confident bounding boxes may consider some latent information that is ignored by the most confident boxes. Suppose an image that a man stretching out his hands, as shown in Figure 5. Some prediction boxes that well catch the upper body or the main body without stretching hands are both inferior detections. However, the average box of these two inferior boxes seems to well catch the entire person. Especially in the case that two boxes have similar confidences, predicting the average box is more convincing than the higher one.

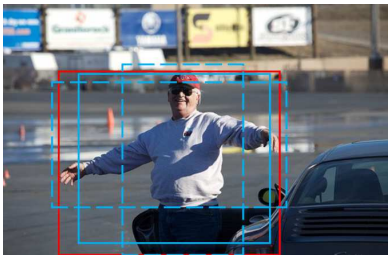

Figure 5. Example image of the mon-maximum weighted method. The red box indicates ground true while blue dotted boxes are predictions. The average box of predictions is shown as the blue solid box.

In our prior work [2], weighted-averaging the non-maximum boxes slightly improve the detection performance without deceleration. The proposed method, named non-maximum weighted (NMW), implements the function:

$$\boldsymbol{b}_{pre} = \frac{\sum_{i=1}^{n} \omega_i \times \boldsymbol{b}_i}{\sum_{i=1}^{n} \omega_i}$$

$$\omega_i = \boldsymbol{c}_i \times iou(\boldsymbol{b}_i, \boldsymbol{b}_{\mathrm{argmax}_i \, \boldsymbol{c}_i})$$

Here $\boldsymbol{b}_i$ is the ith instance in box set $\boldsymbol{B}$ and $\boldsymbol{c}_i$ represents its maximum category confidence. ω_i is the related-confidence for each prediction box and iou function computes jaccard overlap between $\boldsymbol{b}_i$ and the most confident box $\boldsymbol{b}_{\mathrm{argmax}_i \, \boldsymbol{c}_i}$. To obtain these related-confidences, we calculate the product of its own confidence and the overlap with the most confident predictions, which achieve the greatest improvement in our expression experiments. Eventually, the final prediction boxes are generated by calculating the weighted average over box set $\boldsymbol{B}$.

4. Experiments

4.1. Experiment Settings

To construct the proposed framework, some improvements are applied to superior SSD network [16]. Firstly, the pivotal improvement is sharing parameters between Detectors. Different from SSD trains multiple Detectors for multi-scale objects, sharing the parameters makes them practically equivalent to one Detector for arbitrary object sizes. Secondly, maxout is employed in Detector to approximate more complicated curves, while the hyper-parameter k is set to 3. Thirdly, to harmonize the Convolution network with Detector, Feature map towers are adopted as Adapters between the base network and additional convolution layers. Furthermore, we append additional small prior boxes to improve detection performance on small objects by compelling all layers to produce prior boxes with [2, 3] aspect ratios. In addition, instead of traditional NMS in conventional detection frameworks, NMW is applied in our CAD framework to eliminate the redundant predictions.

Table 1. PASCAL VOC2007 test detection results. All networks are trained on the union of VOC2007 trainval and VOC2012 trainval.

Method	network	mAP	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	bike	person	plant	sheep	sofa	train	tv
YOLOv2	Darknet	75.4	86.6	85.0	76.8	61.1	55.5	81.2	78.2	91.8	56.8	79.6	61.7	89.7	86.0	85.0	84.2	51.2	79.4	62.9	84.9	71.0
Faster	Residual-101	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
R-FCN	Residual-101	80.5	79.9	87.2	81.5	72.0	69.8	86.8	88.5	89.8	67.0	88.1	74.5	89.8	90.6	79.9	81.2	53.7	81.8	81.5	85.9	79.9
SSD	VGG	77.2	81.3	85.3	76.6	70.9	50.0	84.3	85.5	88.1	59.0	79.8	76.0	86.1	87.3	84.2	79.4	51.9	77.7	77.7	87.7	75.3
DSSD	Residual-101	78.6	81.9	84.9	80.5	68.4	53.9	85.6	86.2	88.9	61.1	83.5	78.7	86.7	88.7	86.7	79.7	51.7	78.0	80.9	87.2	79.4
I-SSD	VGG	78.6	82.4	84.0	77.4	71.3	52.8	85.7	87.0	88.3	62.6	82.7	78.1	86.7	88.8	86.0	79.9	53.1	77.6	80.8	87.9	77.8
CAD	VGG	79.3	80.7	87.2	78.0	73.0	56.5	86.7	87.0	87.9	62.5	84.0	79.1	85.8	87.1	85.3	80.8	55.7	80.0	81.0	88.6	79.0

Table 2. Comparison of speed and accuracy between some excellent networks. The mAPs are evaluated on PASCAL VOC2007 test.

Method	mAP	FPS	Proposals	GPU	Input size
Faster-RCNN	73.2	7	6000	Titan X	∼1000x600
Faster-RCNN	76.4	2.4	300	K40	∼1000x600
R-FCN	80.5	9	300	Titan X	∼1000x600
YOLO9000	78.6	40	845	Titan X	544x544
SSD	77.2	46	8732	Titan X	300x300
DSSD	78.6	9.5	17080	Titan X	321x321
I-SSD	78.6	16	8732	K20	300x300
CAD	79.3	35	11640	GTX 1070	300x300

To validate the superiority of our framework, comprehensive experiments are conducted on two object detection benchmarks: PASCAL VOC2007 and VOC2012 [29]. We implement the proposed framework by using the Caffe framework on NVIDIA GTX1070 with CUDA 8.0 and cuDNN v5.1. Stochastic gradient descent (SGD) is employed as the optimization algorithm for the proposed framework while batch size is fixed to 32. The proposed framework is trained with 10^{-3} learning rate for the first 80k iterations and decayed as a factor of 0.1 for every 20k iterations while 120k iterations totally. Moreover, to enhance the generalization ability of the proposed framework, we apply data augmentation as [16]. Furthermore, VGG16 [5] is employed as backbone network of the proposed framework to imitate the correlation between the image pixels and corresponding function outputs. In addition, the input image size of our network is set to 300x300 for a fair comparison to the original SSD framework.

4.2. PASCAL VOC2007

PASCAL VOC2007 test set contains 4952 labeled images with RGB channels, including 20 categories of objects with various sizes and positions. To evaluate the proposed framework on VOC2007 test set, our network is trained on VOC2007 trainval and VOC2012 trainval (07+12). As shown in Table 1, experiment results are compared with some excellent frameworks, such as Faster R-CNN, YOLO, R-FCN and SSD.

We have the following observations from the results. Firstly, compared to other end-to-end frameworks, the proposed network significantly surpasses the original SSD network and achieves superior detection performance. Meanwhile, detection speed of the proposed framework is comparable to the original SSD and YOLO9000. Secondly, the proposed network surpasses other improved SSD-based network, such as DSSD [18] and I-SSD [2], on both speed and detection performance. Thirdly, Faster-RCNN, the primal RPN-based detection framework, is completely defeated by the proposed framework too. Furthermore, state-of-the-art R-FCN network overcomes the proposed network on detection accuracy, however, is 4x slower. In addition, the proposed network achieves remarkable improvements on some low accuracy categories, such as plant and boat.

Table 3. Experiment results on PASCAL VOC 2012 test set. All networks are trained by 07++12: 07 trainval + 07 test + 12 trainval. Result link is the detailed detection performance: http://host.robots.ox.ac.uk:8080/anonymous/CYWQES.html.

Method	mAP	FPS	GPU	Input size
Faster	73.8	7	TITAN X	∼1000x600
R-FCN	77.6	9	TITAN X	∼1000x600
YOLOv2	73.4	40	TITAN X	544x544
SSD	75.8	46	TITAN X	300x300
DSSD	76.3	9.5	TITAN X	321x321
CAD	76.9	35	GTX1070	300x300

Besides, detailed comparisons of framework architectures are shown in Table 2. It is noteworthy that the superior performance of the proposed network is achieved with the smallest input size. The state-of-the-art R-FCN network has 6x larger input size than the proposed framework while only 1% mAP improvement. Other frameworks with similar or larger input size all are defeated by the proposed network. This comparison well demonstrates the superiority of the proposed framework.

4.3. PASCAL VOC2012

To validate the conclusions of the proposed framework in the VOC2007 experiments, we additionally evaluate our network on PACSAL VOC2012 test set. The proposed

Table 4. Controlled experiments in our CAD framework. All mAPs are tested on VOC2007 test.

	SSD					CAD
share params		✓	✓	✓	✓	✓
add Adapters			✓	✓	✓	✓
uniform asp				✓	✓	✓
apply maxout					✓	✓
NMW						✓
FPS	39	39	37	36	35	35
mAP	77.2	77.6	78.1	78.5	79.0	79.3

network is trained by using VOC2007 trainval+test and VOC2012 trainval while tested on VOC2012 test set (10991 images). Experiment results are presented in Table 3.

The same performance trend is obtained as we observed on VOC2007 test. The proposed framework defeats most existing detection frameworks, except the R-FCN network, with the smallest input size and real-time speed. With respect to speed and accuracy, as far as our knowledge, the proposed framework achieves the state-of-the-art detection performance. In addition, the remarkable performance further demonstrates the effectiveness of the proposed framework in object detection.

4.4. Component Analysis

To thoroughly investigate improvement of the proposed framework, we carry out controlled experiments for each component. All experiment networks utilize the same settings as on VOC2007 test and the results are shown in Table 4.

As controlled experiments, we train multiple networks by employing different improvement strategies to the former experimental network. Firstly, the parameters between various Detectors in original SSD are shared by separating each Detector into several sub-detectors that only have one aspect ratio. These sub-detectors are gathered and share parameters to other sub-detectors that have the same aspect ratio. Then, they are reassembled into former Detectors. Secondly, feature map towers are employed as Adapters in the proposed framework to coordinate the base network and Detector. Thirdly, we compel all Detectors to produce prior boxes with [2, 3] aspect ratios for the convenience in parameter sharing. Another advantage of this policy is that network produces more prior boxes to match the small target objects. Furthermore, maxout is adopted to enhance the adaptation and generalization abilities of the proposed framework. Finally, the NMW method eliminates the redundant predictions.

4.5. Post-Process Method Comparison

To demonstrate the superiority of non-maximum weighted method, we compare its performance to NMS and Soft-NMS [31], a new method that weakens the confidences of

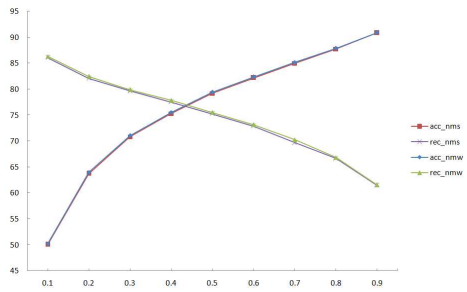

Figure 6. Comparison between NMW and NMS. The x-axis shows the confidence thresholds while the y-axis is the percentages of accuracy or recall.

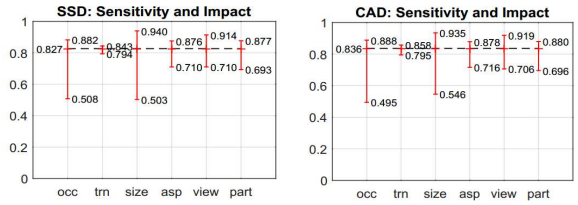

Figure 7. The average (over categories) AP_N performance of the highest performing and lowest performing subsets within each characteristic (occlusion, truncation, bounding box area, aspect ratio, viewpoint, part visibility). Overall AP_N is indicated by the black dashed line. The difference between max and min indicates sensitivity; the difference between max and overall indicates the impact.

non-maximum boxes. Accuracy and recall are the primary criterions to evaluate the effectiveness of these methods. We adjust the confidence threshold, which determines whether the prediction boxes are positive or not, from 0.1 to 0.9 with a step of 0.1. Experiment results are exhibited in Figure 6. The x-axis shows the confidence thresholds while the y-axis is the percentages of accuracy or recall. Unlike superior improvements on RPN-based network, the Soft-NMS have negligible influence to NMS in the proposed framework. Thus, the results of Soft-NMS method are omitted in the figure. Whereas, the proposed method surpasses NMS in both accuracy and recall rate on all threshold conditions.

RPN-based network detects objects among a fixed number of RoIs that generated by the RPN. These networks only generate hundreds of prediction boxes that are deemed to contain target objects. Soft-NMS improves mAP by decreasing confidences of the non-maximum boxes, which generally have low overlaps with each other. These boxes still are predictions with decreased confidences that improve lower threshold accuracy. On the other hand, the proposed framework generates identical prior boxes, whether input images contain target objects are not guaranteed, depend on the image size but not the input image pixels. In addition, due to these boxes have high overlaps with each other, Soft-NMS is likely to allocate 0 or even negative confidences to these boxes. On the contrary, these dense prior

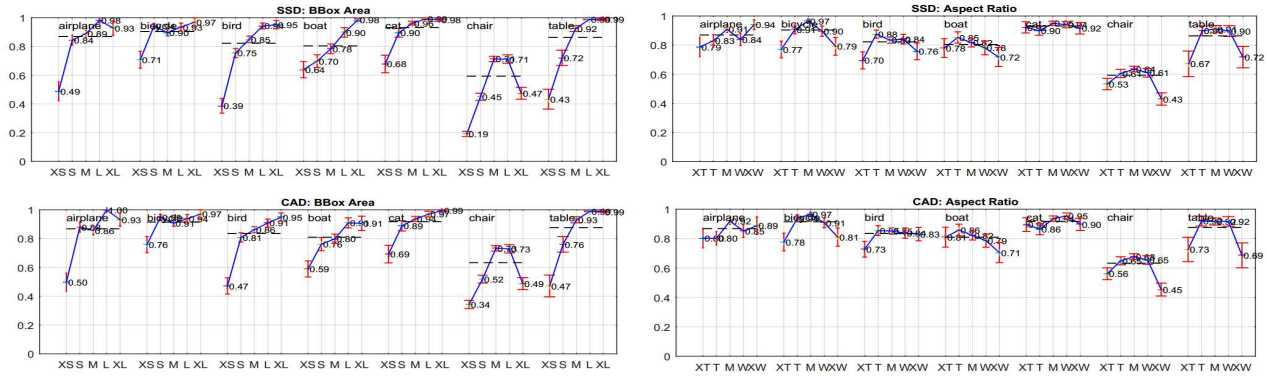

Figure 8. These figures show the impact of object size and aspect ratio on two comparison frameworks: SSD and CAD. Each plot shows the normalized AP with standard error bars (red). Black dashed lines indicate overall normalized AP. Object size is assigned to 5 categories: extra-small (XS: bottom 10%); small (S: next 20%); medium (M: next 40%); large (L: next 20%); extra-large (XL: next 10%). In a similar way, aspect ratio is assigned to other 5 categories: extra-tall/narrow(XT); tall(T); medium(M); wide(W); extra-wide(XW) [30].

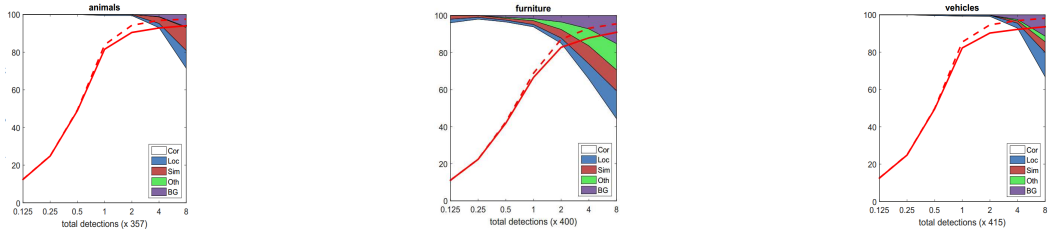

Figure 9. Cumulative fraction of detections that are correct (Cor) or false positive due to poor localization (Loc), confusion with similar categories (Sim), with others (Oth), or with background (BG). The solid red line reflects the change of recall with strong criteria (0.5 jaccard overlap) as the number of detections increases. The dashed red line is using the weak criteria (0.1 jaccard overlap) [16].

boxes are treated in a comprehensive manner that these boxes are considered as supplements in our novel method. The most confident box absorbs these supplements and integrated to a more inclusive box. However, this method may not achieve substantial improvements in RPN-based framework due to the independence of RoIs.

4.6. Invariance Analysis

Furthermore, we utilize the detection analysis tool from [30] to understand the improvements of the proposed framework better. Overall comparisons of the proposed framework and SSD are shown in Figure 7. In addition, the impacts of the object size and the aspect ratio are shown in Figure 8. For all figures, the y-axis indicates the normalized precision (AP_N), defined in [30].

First of all, the proposed framework achieves remarkable AP_N performance and surpasses the SSD network. Secondly, the proposed framework substantially reduces the influences of object size. Furthermore, for different aspect ratios, our framework has more stable detection performance than the SSD network. In addition, target categories are divided into some sets base on their semantics. Two categories are considered to be semantically similar if they are both within one of these sets: {all vehicles}, {all animals including person}, {chair, diningtable, sofa}, {aeroplane, bird}. Detection results of these sets are visualized in Figure 9. Compared to SSD network, our framework reduces all types of error rates, especially the location error.

5. Conclusion

In this work, we present a novel CAD framework to detect target objects in vision range. Three components (Convolution network, Adapters and Detector) are integrated into a unified network to achieve real-time detection speed. By training Detector using multi-scale image patches, the proposed framework is more robust to scale variance. Moreover, maxout and NMW are employed to enhance generalization ability of our framework. Extensive experiments demonstrate that the proposed framework achieves superior performance with respect to speed and accuracy.

6. Acknowledgment

This work was partially supported by the National Key Research and Development Program of China under Grant 2016YFB1001001, the National Natural Science Foundation of China (Grant No. 61522203, 61772275 and 61732007) and the Natural Science Foundation of Jiangsu Province (Grant BK20140058 and BK20170033).

References

[1] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. C. Courville, and Y. Bengio. Maxout networks. In *ICML*, 2013.

[2] C. Ning, H. Zhou, Y. Song, and J. Tang. Inception single shot multibox detector for object detection. In *ICME*, 2017.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[4] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, 2015.

[5] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[6] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[7] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Computer Vision and Pattern Recognition*, 2014.

[8] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37:1904–1916, 2014.

[9] S. Ren, K. He, R. B. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[10] S. Bell, C. L. Zitnick, K. Bala, and R. B. Girshick. Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2874–2883, 2016.

[11] S. Gidaris and N. Komodakis. Locnet: Improving localization accuracy for object detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 789–798, 2016.

[12] T. Kong, A. Yao, Y. Chen, and F. Sun. Hypernet: Towards accurate region proposal generation and joint object detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 845–853, 2016.

[13] M. Najibi, M. Rastegari, and L. S. Davis. G-cnn: An iterative grid based object detector. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2369–2377, 2016.

[14] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick. Mask r-cnn. *CoRR*, abs/1703.06870, 2017.

[15] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[16] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *ECCV*, 2016.

[17] J. Redmon and A. Farhadi. Yolo9000: Better, faster, stronger. *CoRR*, abs/1612.08242, 2016.

[18] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg. Dssd : Deconvolutional single shot detector. *CoRR*, abs/1701.06659, 2017.

[19] P. F. Felzenszwalb, D. A. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[20] Z. Li and J. Tang. Weakly supervised deep matrix factorization for social image understanding. *IEEE Trans. Image Processing*, 26(1):276–288, 2017.

[21] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

[22] M. Lin, Q. Chen, and S. Yan. Network in network. In *ICLR*, 2014.

[23] Z. Li, J. Liu, J. Tang, and H. Lu. Robust structured subspace learning for data representation. *IEEE Trans. Pattern Anal. Mach. Intell*, 37(10):2085–2098, 2015.

[24] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 104:154–171, 2013.

[25] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov. Scalable object detection using deep neural networks. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2155–2162, 2014.

[26] R. B. Girshick. Fast r-cnn. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448, 2015.

[27] J. Dai, Y. Li, K. He, and J. Sun. R-fcn: Object detection via region-based fully convolutional networks. In *NIPS*, 2016.

[28] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *CoRR*, abs/1312.6229, 2013.

[29] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010.

[30] D. Hoiem, Y. Chodpathumwan, and Q. Dai. Diagnosing error in object detectors. In *ECCV*, 2012.

[31] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis. Improving object detection with one line of code. 2017.