Multi-Task Vehicle Detection With Region-of-Interest Voting

Wenqing Chu, Yao Liu, Chen Shen, Deng Cai, *Member, IEEE*, and Xian-Sheng Hua, *Fellow, IEEE*

***Abstract*— Vehicle detection is a challenging problem in autonomous driving systems, due to its large structural and appearance variations. In this paper, we propose a novel vehicle detection scheme based on multi-task deep convolutional neural networks (CNNs) and region-of-interest (RoI) voting. In the design of CNN architecture, we enrich the supervised information with subcategory, region overlap, bounding-box regression, and category of each training RoI as a multi-task learning framework. This design allows the CNN model to share visual knowledge among different vehicle attributes simultaneously, and thus, detection robustness can be effectively improved. In addition, most existing methods consider each RoI independently, ignoring the clues from its neighboring RoIs. In our approach, we utilize the CNN model to predict the offset direction of each RoI boundary toward the corresponding ground truth. Then, each RoI can vote those suitable adjacent bounding boxes, which are consistent with this additional information. The voting results are combined with the score of each RoI itself to find a more accurate location from a large number of candidates. Experimental results on the real-world computer vision benchmarks KITTI and the PASCAL2007 vehicle data set show that our approach achieves superior performance in vehicle detection compared with other existing published works.**

***Index Terms*— Vehicle detection, CNN, multi-task, region-of-interest.**

I. Introduction

VEHICLE detection is a fundamental problem of many visual computing applications, including traffic monitoring and intelligent driving. Unfortunately, vehicle detection is very challenging due to the large intra-class variations caused by different viewpoints, occlusions and truncations. Figure 1 shows a few examples with varying complexities from the PASCAL2007 car dataset [1] and the recently proposed KITTI vehicle detection benchmark [2].

In general, vehicle detection can be viewed as a special topic of generic object detection. In the past few years, researchers have made remarkable progress to boost the performance of object detection [3]–[8]. A common pipeline to address this problem consists of two main steps: (1) object proposal generation, (2) class-specific scoring and bounding box regression. For the first step, there is a significant body of well-designed methods [8]–[11] for generating object proposals or just a sliding window fashion employed in [5]. Then some specific visual features of the object bounding box are extracted and a classifier is utilized to determine whether the bounded area is a desired object or not, in which the representative methods include AdaBoost algorithm [3], DPM models [5] and deep CNN models [7]. However, vehicle detection is still challenging due to its large structural and appearance variations, especially ubiquitous occlusions which further increase the intra-class variations. In addition, many vehicle detection benchmarks require the Intersection over Union (IoU) surpassing 0.7 to assess a correct localization, which lifts the performance requirement significantly.

In this paper, we propose a novel vehicle detection scheme based on multi-task deep convolutional neural networks (CNN), region-of-interest (RoI) voting and multi-level localization, denoted by RV-CNN. Multi-task learning is designed to impose knowledge sharing while solving multiple correlated tasks simultaneously, boosting the performance of a part or even all of the tasks [12]. In our method, the CNN model is trained on four tasks: category classification, bounding box regression, overlap prediction and subcategory classification. Here, we introduce the subcategory classification task to enforce the CNN model to learn a good representation for vehicles under different occlusions, truncations and viewpoints. We utilize the proposed concept of 3D Voxel Pattern (3DVP) in [13] for subcategory classification. 3DVP is a kind of object representation that jointly captures key object properties which relates appearance, object pose, occlusion and truncation for rigid objects in the clustering process. Then each 3DVP is considered to be a subcategory.

Most detection methods employ the prediction scores from the CNN model to perform non-maximum suppression (NMS) to get the final bounding box locations. However, detection scores above a certain level are not strongly related to the reliability of box proposals [14]. One reason is that the

Manuscript received April 26, 2017; revised September 3, 2017 and October 5, 2017; accepted October 6, 2017. Date of publication October 12, 2017; date of current version November 3, 2017. This work was supported in part by the National Basic Research Program of China (973 Program) under Grant 2013CB336500 and in part by the National Youth Top-notch Talent Support Program. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Tolga Tasdizen. *(Corresponding author: Deng Cai.)*

W. Chu and Y. Liu were with Alibaba, Hangzhou 311121, China. They are now with the State Key Laboratory of CAD&CG, College of Computer Science, Zhejiang University, Hangzhou 310027, China (e-mail: wqchu16@gmail.com; xuanyao0111@gmail.com).

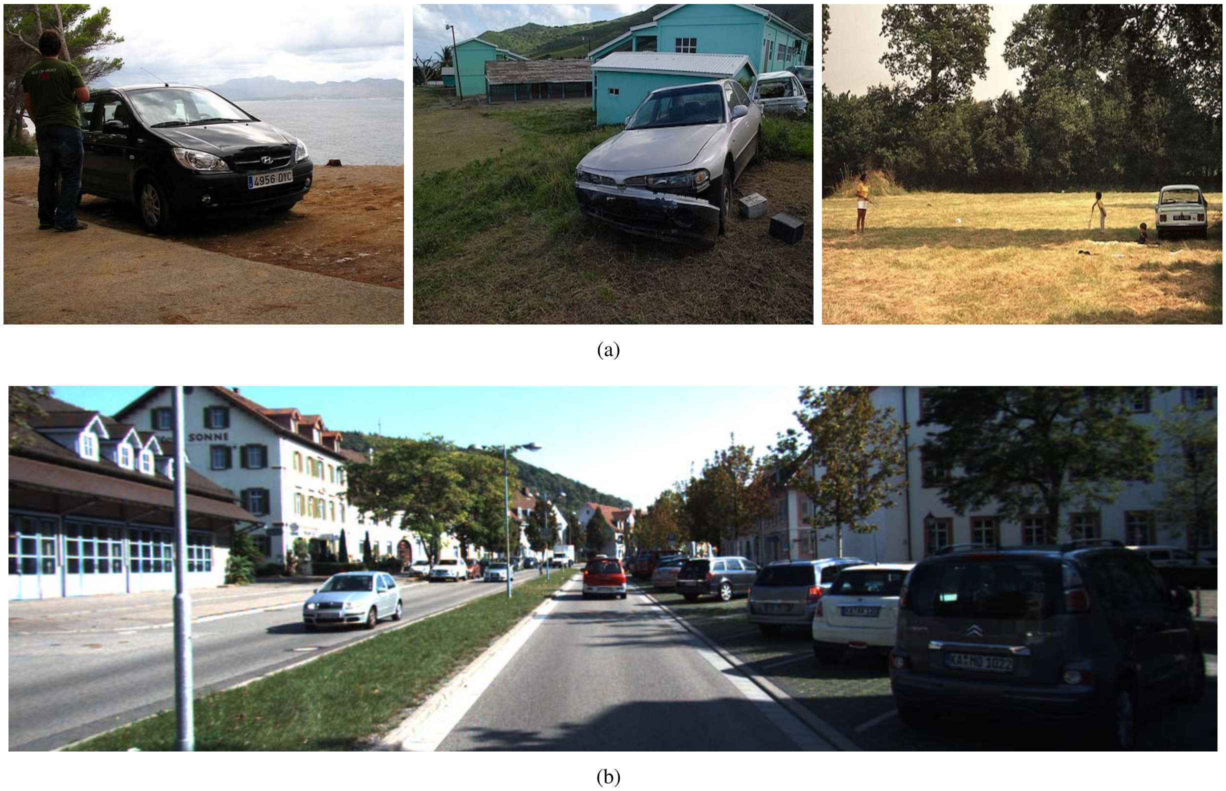
C. Shen was with Alibaba, Hangzhou 311121, China. He is now with the School of Instrumental Engineering, Zhejiang University, Hangzhou 310027, China (e-mail: zjushenchen@gmail.com).

D. Cai is with the State Key Laboratory of CAD&CG, College of Computer Science, Zhejiang University, Hangzhou 310027, China (e-mail: dengcai@gmail.com).

X.-S. Hua is with the Alibaba Group, Hangzhou 311121, China (e-mail: xiansheng.hxs@alibaba-inc.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2017.2762591



(a)

(b)

Fig. 1. Illustration of varying complexities in vehicle detection from two datasets. (a) The PASCAL VOC2007 car dataset [1] consists of single cars under different viewpoints but with less occlusion. (b) The KITTI vehicle benchmark [2] includes on-road cars captured by a camera mounted upon a driving car which have more occlusions and truncations.

classifier is trained to classify objects from background rather than ranking the intersection-over-union (IoU). Therefore, we propose to use neighboring RoIs to refine this score. First, we employ the CNN model to predict the offset direction from the RoI towards the ground truth for each boundary simultaneously. With these additional information, we design a simple yet effective voting scheme to rescore those RoIs. After the scores of all proposals are calculated again, we can apply NMS to get final results. Furthermore, in our observation, the output from region proposal network [8] is not guaranteed to reach 100% recall under the constraint that IoU should surpass 0.7. That will pose a challenge to the following detection network because it has to tackle some difficult cases without high quality proposals. Also, in Faster R-CNN [8] the detection scores of the predicted boxes for NMS is not accurate because it applies the conv feature of the RoI before regression. Taking these two drawbacks into account, a multi-level localization scheme is also explored to further improve detection accuracy and reliability.

We have evaluated our method on two commonly used vehicle detection datasets (the KITTI vehicle benchmark [2] and the PASCAL VOC2007 car dataset [1]). Our method achieves 91.67% Ap on the KITTI vehicle detection benchmark, surpassing recent results [15]–[17] by notable margins. In addition, we also conduct experiments on PASCAL VOC2007 car dataset. Experimental results show a consistent and significant performance gain with our RV-CNN model compared with baseline and related methods.

II. Related Work

In this section, we briefly review the recent work on general object detection and vehicle detection.

Generic object detection is an active research area in recent years, resulting in a large amount of prior works. One of the earliest methods to achieve real-time detection with relatively high accuracy is the cascaded detector in [3]. This architecture has been widely used to implement sliding window detectors for faces [3], [18], pedestrians [19] and vehicles [20]. Part-based model is also one of the most powerful object detection approaches in the literature, in which the deformable part-based model (DPM) [5], [21] is an excellent example. This method takes the histogram of oriented gradients (HOG) features as input and utilizes a star-structured architecture consisting of root and parts filters to represent highly variable objects which made it capable of detecting objects with severe occlusion.

Recently, deep convolutional neural networks (CNN) have demonstrated superior performance, dominating the top accuracy benchmarks in various vision tasks [22]–[26]. These works inspired a significant body of methods [7], [8], [27]–[36] addressing the problem with CNN models. Among these approaches, the regions-with-convolutional-neural-network (R-CNN) framework [7] has achieved promising detection performance and became a commonly employed paradigm for object detection. Its essential steps include object proposal generation with selective search [9], CNN features extraction, object candidates classification and regression based on the CNN features. However, R-CNN brings excessive

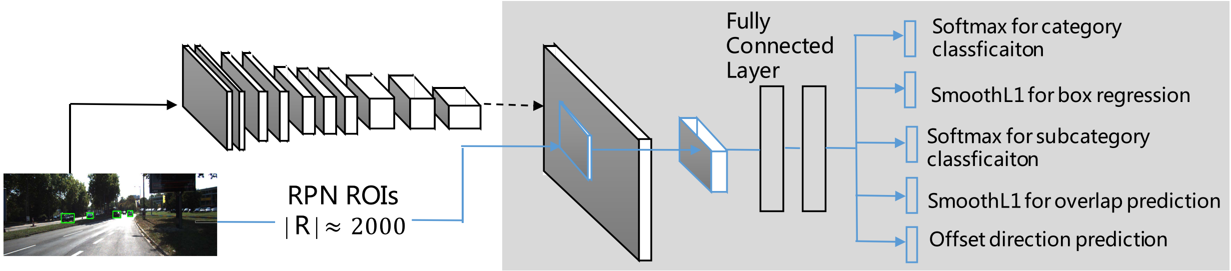

RPN ROIs
|R| ≈ 2000
Fully Connected Layer
Softmax for category classficaiton
SmoothL1 for box regression
Softmax for subcategory classficaiton
SmoothL1 for overlap prediction
Offset direction prediction

Fig. 2. Illustration of multi-task framework.

computation cost because it extracts CNN features repeatedly for thousands of object proposals. Spatial pyramid pooling network (SPPnet) [28] and Fast Region-based Convolutional Network (Fast R-CNN) [29] were proposed to accelerate the process of feature extraction in R-CNN by sharing the forward pass computation. The drawback of them is that they still employ bottom-up proposal generation which is the bottleneck of efficiency. Instead, the authors proposed a Region Proposal Network (RPN) method [8] that shares full-image convolutional features with the detection network, thus enabling nearly cost-free region proposals. The MS-CNN [15] consists of a proposal sub-network and a detection sub-network. In the proposal sub-network, detection is performed at multiple output layers, so that receptive fields match objects of different scales. This scheme is also utilized in SSD [32] and TextBoxes [37]. Another interesting work is YOLO [31], which outputs object detections within a 7x7 grid. This network runs at 40 fps, but with some compromise of detection accuracy.

Most of these deep models target general object detection. To better handle the detection problem of the occluded vehicles, a second-layer conditional random field (CRF) was used over root and part score configurations provided by a DPM model in [38]. Recently, an AND-OR structure was proposed in [39] and [40] to model occlusion configurations effectively compared against the classical DPM. In [41], the authors proposed to combine vehicle detection and attributes annotation. In addition, a common approach for improving model generalization is to learn subcategories within an object class [20]. Subcategory has been widely utilized to facilitate vehicle detection, and several methods of subcategory classification have been proposed [42]–[45]. In [42], visual subcategories corresponding to vehicle orientation were learned in an unsupervised manner using Locally Linear Embedding and HOG features. Reference [43] performs clustering according to the viewpoint of the object to discover subcategories. Discriminative subcategorization, where the clustering step considers negative instances, was studied in [45]. More recently, [13] proposed a novel object representation, 3D Voxel Pattern (3DVP), that jointly encodes the key properties of objects including appearance, 3D shape, viewpoint, occlusion and truncation. The method discovers 3DVPs in a data-driven way, and train a bank of specialized detectors for a dictionary of 3DVPs. In [46], the authors utilized the 3DVP subcategory information to train the subcategory conv layer outputs heat maps about the presence of certain subcategories at a specific location and scale. In our work, we employ the subcategory classification as part of the multi-task to improve CNN-based detection performance and this component can be implemented with the subcategory labels obtained in [13], [42], and [43].

III. Multi-Task Vehicle Detection With RoI Voting

In this section, we describe our multi-task deep convolutional neural networks for solving the vehicle detection problem. For each input image, our approach consists of three main stages. Firstly, we generate a pool of object proposals obtained by multi-scale Region Proposal Network (RPN) [8]. Then we use a multi-task CNN model to predict the attributes of each RoI. According to the regression results, some proposals will be processed by a second level regression network. Finally, we employ an efficient voting mechanism to refine the final score of each RoI. In addition, since we can obtain the subcategory information, we introduce the subcategory-aware non-maximum suppression (NMS) to tackle the occlusions better. Finally, we can obtain predicted boxes which are very accurate for real applications.

A. Multi-Task Loss Function

Recently, multi-task learning has been applied to many computer vision problems, particularly when there is a lack of training samples [12]. Multi-task learning is intended to impose knowledge sharing while solving multiple correlated tasks simultaneously. It has been demonstrated that this sharing can boost the performance of a part or all of the tasks [12], [47], [48]. For the vehicle detection problem, we enrich the supervised information with subcategory, region overlap, bounding-box regression and category of each training RoI as a multi-task learning framework. Next, we will explain the details of the proposed approach of the multi-task CNN model. Figure 2 shows the overall flow of the proposed multi-task learning framework. As illustrated in figure 2, after generating RoIs, we apply the RoI pooling layer proposed in [29] to pool conv features for each RoI. Then the pooled conv features are used for accomplishing four tasks: category classification, bounding box regression, overlap prediction and subcategory classification. The last part "offset direction prediction" will be described in next section.

Each training RoI is labeled with a ground-truth class and a ground-truth bounding-box regression target, which is similar to the setting in [29]. Generally, this supervised information are utilized to design the classification loss L_{cat} and bounding-box regression loss L_{loc}.

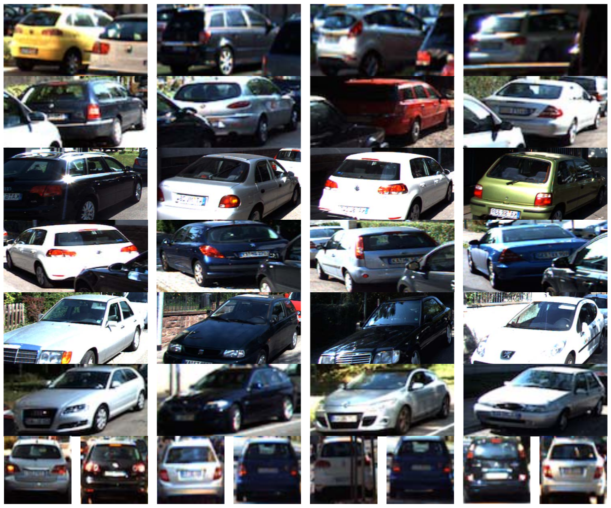


Fig. 3. Each row is a subcategory.

Next, the third task is subcategory classification. For vehicle detection within complex and cluttered urban scenes, occlusion and viewpoint are critical aspects. As in [40], handling occlusions requires models capable of capturing the underlying regularities of occlusions at part level (i.e. different occlusion configurations) and explicitly exploit contextual information co-occurring with occlusions, which goes beyond single-vehicle detection. In addition, 2D images in different views are also hard to recognize. These increase the intra-class variations significantly. To represent both occlusion and viewpoint variations, we adopt the concept of 3D Voxel Pattern (3DVP), which is proposed recently in [13]. A 3DVP is an object representation that jointly captures key object properties which relates to appearance, 3D shape and occlusion masks. Reference [13] proposed to utilize 3D CAD models in repositories on the web, such as the Trimble 3D Warehouse, and register these 3D CAD models with 2D images to build 3D voxel exemplars. To be more specifically, for each image in the training set, an object in the image is registered with a 3D CAD model selected from a pre-defined collection of models, where the model which has the closest aspect ratio with the ground truth 3D cuboid of the object instance is selected. Then all the registered 3D CAD models are projected onto the image plane using the camera parameters and obtain the depth ordering mask. In the following, the depth ordering mask determines which pixel of the projected 3D CAD model is visible, occluded, or truncated. A 3DVP represents a group of 3D voxel exemplars which share similar visibility patterns encoded in their 3D voxel models. Reference [13] discovers 3DVPs by clustering 3D voxel exemplars in a uniform 3D space. For more details, the readers can refer to their project website.[1]

Following [13], we employ the 3D Voxel Pattern (3DVP) representation for rigid objects (i.e., vehicle in KITTI), which jointly models object pose, occlusion and truncation in the clustering process. Then each 3DVP is considered to be a subcategory. Figure 3 shows a few examples with varying subcategories of vehicles from KITTI vehicle datasets. With these additional annotations, the CNN model can capture more key information for detecting. As illustrated in figure 2, the CNN model outputs a discrete probability distribution (per RoI), $p = (p_0, \cdots, p_K)$, over $K+1$ subcategories. As usual, p is computed by a softmax over the $K+1$ outputs of a fully connected layer. Therefore, the loss for subcategory classification is formulated as $L_{sub}(p,u) = logp_u$ which is log loss for true class u.

In addition, we find that predicting the overlap between the RoI and the corresponding ground truth is beneficial for other tasks. For overlap regression, we use the loss in Eq 1.

$$L_{iou}(O_p, O_g) = smooth_{L1}(O_p - O_g) \tag{1}$$

in which

$$smooth_{L1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \ , \\ |x| - 0.5 & \text{otherwise.} \end{cases}$$

is a robust $L1$ loss that is less sensitive to outliers than the $L2$ loss, which requires careful tuning of learning rates in order to prevent exploding gradients. The O_p denotes the overlap predicted by the CNN model and the O_g is computed according to the ROI and the ground truth.

In summary, the loss of the whole multi-task framework can be formulated as:

$$L_{multi-task} = L_{cat} + \lambda_1 L_{loc} + \lambda_2 L_{iou} + \lambda_3 L_{sub} \tag{2}$$

The hyper-parameter $\lambda_1, \lambda_2, \lambda_3$ in Eq 2 are used to control the balance between the four task losses. We tuned these hyper-parameters on the validation dataset. Specifically, the λ_1, λ_2 and λ_3 were set to 1, 10, 1.2 in the experiments.

B. Region of Interest Voting

It is observed that the detection scores cannot well represent the reliability or confidence of the bounded area. In [14], the authors also argued that detection scores above a certain level are not strongly related to the optimality of box proposals. Actually this is not surprising because the classifiers are trained to classify objects from background rather than ranking the IoU. In addition, the scores of the predicted boxes are computed by the conv feature of RoIs which are slightly different from the regressed boxes which is also questionable. To handle this problem, we use the neighboring RoIs to refine its score. First, we employ the CNN model to predict the offset direction from each RoI boundary towards the ground truth boundary simultaneously. Then we can get four variables which indicate the directions towards the ground truth. In our method, we denote these four variables with D_l, D_t, D_r, D_d for left boundary, top boundary, right boundary and down boundary of the RoI respectively. For example, possible predictions for D_l are the following: “go to left”, “go to right”, “stop here” and “no instance around this RoI”. And for D_t, “go to up”, “go to down”, “stop here” and “no instance around this RoI” are possible training labels. These labels can be computed according to the positions of the ROI and the ground truth before training.

As aforementioned, we use a multi-scale RPN model to generate thousands of object proposals. With the proposed

[1] http://cvgl.stanford.edu/projects/3DVP/

multi-task CNN framework, the bounding box offsets, scores and directions are predicted for each RoI. Then combining the coordinates of each RoI and the corresponding box offsets, we can get a large number of predicted boxes, which is much larger than the actual number of objects in an image. Consequently, we divide all predicted boxes in one image into groups, each corresponding to one object. The grouping scheme is simply as following: we select the predicted boxes with the highest score as the seed, and put boxes that have high IoUs with the seed into one group. This process iterates until all boxes are assigned. This scheme is common in object detection [5], [7], [8], [29]. We aim to find the optimal object prediction box for each group. Previous methods select the predicted boxes with the highest prediction score directly. Here, we utilize the additional information from neighboring RoIs of each predicted box to refine the score. If the position of the predicted box agrees with the predicted direction of its neighboring RoI, then this predicted box is more reliable. Otherwise, the final score of the predicted box should be decreased. For clarity, suppose a predicted box has coordinates $b = \{x_1, y_1, x_2, y_2\}$ and score s. And we denote its neighboring RoIs by B, the number of RoIs in B by N and the i-th RoI with assigned score s_i and predicted directions D_l^i, D_t^i, D_r^i, D_d^i by $b_i = \{x_1^i, y_1^i, x_2^i, y_2^i\}$. Then we formulate the voting scheme as in 3.

$$s^{'} = s + \lambda \sum_{b\in\{l,t,r,d\}} \sum_{i=1}^{N} R_b(b, b_i) \tag{3}$$

in which

$$R_l(b, b_i) = \begin{cases} s_i & \text{if } x_1 < x_1^i \text{ and } D_l^i = \text{“go to left” }, \\ -s_i & \text{if } x_1 < x_1^i \text{ and } D_l^i = \text{“go to right”}, \\ -s_i & \text{if } x_1 > x_1^i \text{ and } D_l^i = \text{“go to left”}, \\ s_i & \text{if } x_1 > x_1^i \text{ and } D_l^i = \text{“go to right”}. \end{cases}$$

Other $R_b(b, b_i)$ functions follow the same rule as $R_l(b, b_j)$. After the scores of all predicted boxes are computed again, we can apply NMS to get final results.

This RoI Voting method has several advantages. First, different from the category classifiers which are trained to classify objects from background rather than ranking the IoU, our RoI voting method predicts the offset directions towards the ground truth, which are sensible to the locations. Furthermore, this RoI Voting method utilizes statistical information from the neighboring RoIs which makes the results more robust and reliable. Second, compared with previous methods solving the detection problem by a CNN-based regression task, our method adopts a more robust classification model which is simple yet effective. A CNN model generally achieves better performance on classification tasks than regression tasks [49]. As the classification of offset directions with soft-max loss forces the model to be maximally activated at a true direction rather than exact values of a bounding box coordinates. In addition, predicting the directions towards the ground truth can be implemented as part of the multi-task framework which does not bring much extra burden in computation.

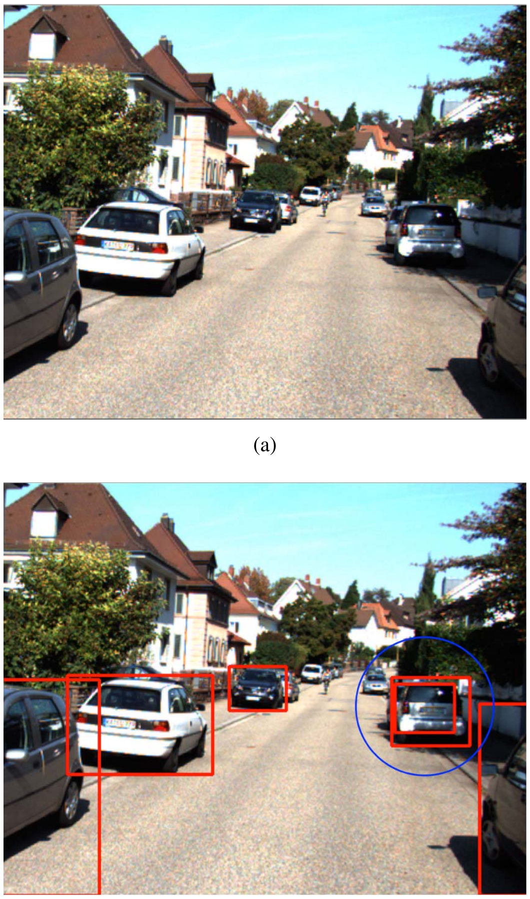
C. Multi-Level Localization

In the common object detection pipeline such as Fast RCNN [29], we find two drawbacks. First, since many detection benchmarks require IoU surpassing 0.7 to assess a correct localization, the region proposal network [8] often fails to reach 100% recall. That will pose a challenge to the following detection network because it has to tackle some difficult cases without high quality proposals. Second, in Fast R-CNN the scores of proposals which are used to do NMS are not accurate because they employ the features before regression. These two factors will decrease the performance of these detectors in practical vehicle detection task. Therefore, we introduce a multi-level localization framework to address these two problems in a coarse-to-fine fashion. Specifically, our localization scheme starts from the region proposal network [8] and works by iteratively scoring them and refining their coordinates. Here, we implement a two-stage scheme. First, we label all proposals which have overlap with ground truth greater than 0.5 as positive samples for training the first-stage regression network. Since we find the RPN fails to recall all vehicles when using 0.7 directly while with 0.5 all vehicles have positive proposals. In the test phase, this regression network can raise up the recall from 97.8% to 98.9%. In the second stage, we use the predicted bounding boxes from first stage to train a second-level object detection network using proposals which have overlap with ground truth greater than 0.7 as positive samples. In this stage, most vehicles have high quality proposals which makes the regression task relatively easy. In addition, we find the outputs of networks at the first level provide robust proposals which makes the second network generate more accurate localization. Furthermore, the bounding box offset computed by the second network is often small which make the scores of the predicted boxes more accurate.

Taking the speed into consideration, we do first level localization for all proposals and select part of them to perform second level localization. The rule for selecting is that: if a proposal has large overlap with the predicted box, we will not do localization for the second time. We believe if the overlap is great, the score is accurate and the proposal does not need regression again. In the experiment part, we set this threshold to 0.9. After the multi-level localization, we obtain a collection of detection results that both exhibit high recall and accurate localization. We considered reusing the conv layers feature to do multi-level localization. However, the performance gain is not satisfactory. Therefore, for the second stage, we train a new regression network. We employ this design because we want the classification scores of the proposals are computed accurately as soon as possible by the conv feature of the corresponding bounding boxes.

D. Subcategory-Aware NMS

In complex traffic scene, the occlusions make the vehicle detection very challenging. For example, there are two cars close to each other in the blue circle in Figure 4 and their IOU is greater than 0.7. Although our previous pipeline can detect their locations and assign them high scores, the standard post-processing step NMS will filter one of the bounding



(a)

(b)

Fig. 4. In the complex traffic scene, the standard NMS will lead to miss detections. (a) Original image. (b) Part of the vehicle detection results before NMS.

box which has lower score. If we set the threshold of the NMS higher, then the two bounding boxes maybe preserved. However, the precision of the detection results will be very low. To handle this dilemma, we introduce the subcategory-aware NMS (subNMS) method. In our multi-task framework, we can obtain the subcategory information. Since the two cars in the blue circle belong to different subcategory, our subNMS exploits a cascade pipeline. First, we perform the standard NMS for the bounding boxes belonging to the same subcategory with a rigorous threshold like 0.5. Then all bounding boxes will be processed by the NMS with a high threshold like 0.75. With the proposed subNMS, the precision and recall of the detection results can achieve a balance.

E. Implementation Details

Our framework is implemented using Caffe [50] and runs on a workstation configured with an NVIDIA M40 GPU card. Instead of training our RPN and detection CNN from scratch, we apply the model pretrained on ImageNet [22] to initialize the conv layers and first two FC layers and finetune the whole network afterwards. On KITTI benchmark, we finetune the AlexNet [22] for the first level localization and the GoogleNet [51] for the second level localization. To tackle the variation of scales, we use a multi-scale way for training the first level localization. Due to GPU memory constraints, we cannot train a multi-scale GoogleNet detection network directly. Therefore, we crop and resize the RoIs independently without sharing convolution computing in the same input image. The fully connected layers used for multi-task learning are initialized from zero-mean Gaussian distributions with standard deviations 0.01 and 0.001, respectively. Biases are initialized to 0. All layers use a per-layer learning rate of 1 for weights and 2 for biases and a global learning rate of 0.001. When training on KITTI train dataset we run SGD for 30k mini-batch iterations, and then lower the learning rate to 0.0001 and train for another 10k iterations. Learning stops after 40,000 iterations and the parameters of layers conv1-1 to conv2-2 are fixed during learning for faster training. When training on VOC07 trainval car dataset we run SGD for 8k mini-batch iterations, and then lower the learning rate to 0.0001 and train for another 2k iterations. A momentum of 0.9 and parameter decay of 0.0005 (on weights and biases) are used.

IV. Experiments

In this section, we evaluate our method on two public datasets: the KITTI vehicle detection benchmark [2] and the PASCAL VOC2007 car dataset [1].

A. Experiments on KITTI Validation Set

The KITTI dataset is composed of 7481 training images and 7518 test images. The total number of objects in training sums up to 51867, in which cars only account for 28742. The key difficulty of KITTI car detection task is that a large number of cars are in small size (height $<$ 40 pixels) and occluded. Since the ground truth annotations of the KITTI testing set is not publicly available, we use the training/validation split of [46] to conduct analyses about our framework, which contain 3682 images and 3799 images respectively. For validation on KITTI, we use 125 subcategories (125 3DVPs for car), while for testing on KITTI, we use 227 subcategories (227 3DVPs for car). Regarding the number of subcategories, we follow the configuration in [13]. The 3DVP is a data-driven method and the number of subcategories is a hyper-parameter used in the clustering algorithm. For the validation dataset, only the training dataset is utilized to discover the 3DVP patterns. And for the testing dataset, the union of the training dataset and the validation dataset is more complex. So the number of subcategories is larger.

We evaluate our recognition results at three difficulty levels, easy, moderate, and hard, suggested by the KITTI benchmark [2]. To evaluate the object detection accuracy, the Average Precision (AP) is reported throughout the experiments. 0.7 overlap threshold is adopted in the KITTI benchmark for car. Table I shows the detection results on the three categories, where we demonstrate the effects of various components on RV-CNN performance on KITTI. From table I, the components of multi-task learning, RoI Voting and multi-level localization are all effective design. For those moderate and hard level cars, our method can achieve better performance with more components. To show the robustness of our method, we give

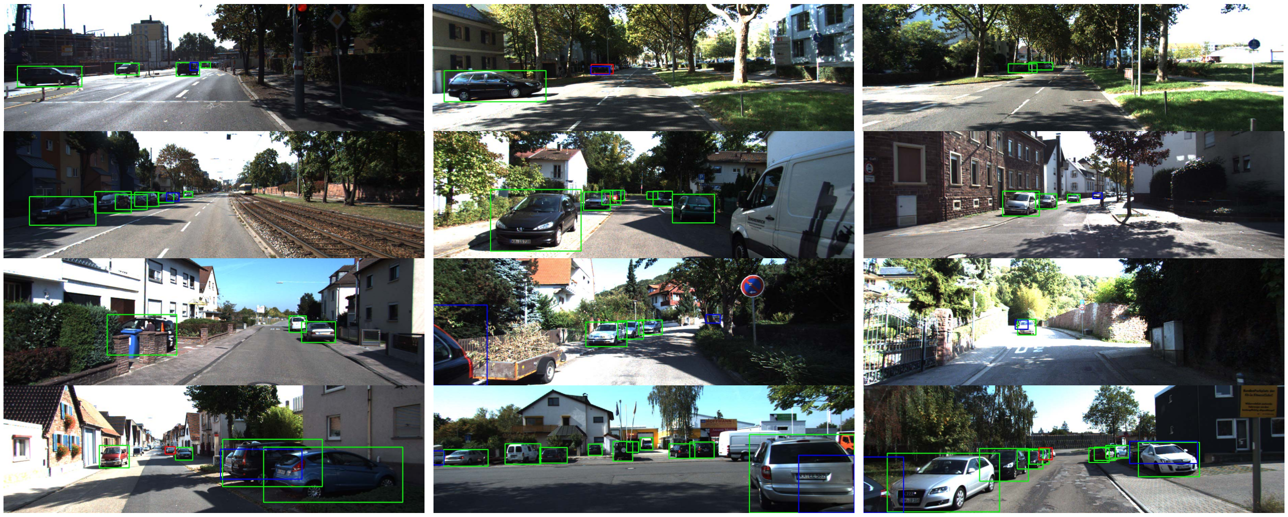


Fig. 5. Examples of successful and failure cases for detection (a green box denotes correct localization, a red box denotes false alarm and a blue box denotes missing detection).

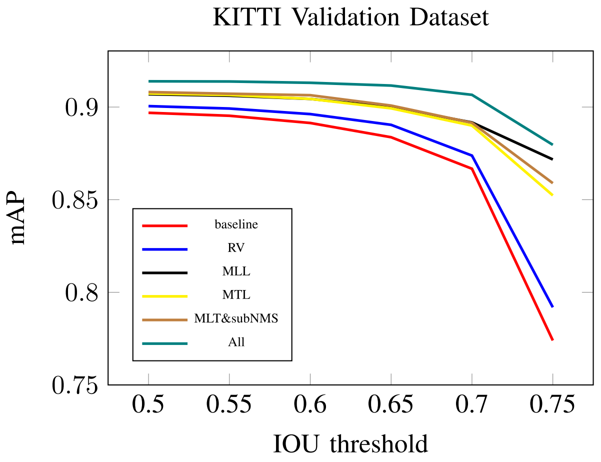


Fig. 6. AP curves under different IOU threshold on the KITTI validation set of the moderately difficult results.

TABLE I

EFFECTS OF VARIOUS COMPONENTS ON RV-CNN PERFORMANCE. "OVERLAP PREDICTION" INDICATES BASELINE AND THE OVERLAP PREDICTION TASK WITHOUT SUBCATEGORY CLASSIFICATION, "MTL" INDICATES MULTI-TASK LEARNING, "RV" INDICATES RoI VOTING AND "MLL" INDICATES MULTI-LEVEL LOCALIZATION. 'ALL' INDICATES ALL COMPONENTS ARE USED

method	Easy	Moderate	Hard
baseline	87.33	86.67	76.78
Overlap prediction	88.22	87.23	77.42
RV	88.37	87.38	77.47
MLL	90.97	89.17	77.94
MTL	90.52	89.00	78.78
MTL & subNMS	90.66	89.15	79.27
All	90.88	90.66	84.33

the AP under different IOU threshold in Fig 6. In addition, fig 5 presents some examples of our detection results on KITTI validation dataset. We can see that those occluded cars which are difficult to see occupy a large part of failure cases for detection. In the future, we need to combine the CNN model with some occlusion reasoning mechanism to handle these difficult cases better.

B. Experiments on KITTI Test Set

To compare with the state-of-the-art methods on the KITTI detection benchmark, we train our RPN and RV-CNN with all the KITTI training data, and then test our method on the KITTI test set by submitting our results to the official website. Table II presents the detection results on the three categories, where we compare our method (RV-CNN) with different methods evaluated on KITTI. These results were extracted in March, 2017. Recently, the evaluation script has been changed and the original results are provided in.[2] Our method ranks on top among all the published methods based on the moderately difficult results. The experimental results demonstrate the ability of our CNNs in handling difficult cars which have more occlusions and truncations. Fig 7 presents precision-recall curves on the KITTI test set of the moderately category.

C. Experiments on VOC Pascal 2007 Car dataet

We also compare our method on another public dataset: the PASCAL VOC2007 car dataset [1] with several competitive models: DPM [5], RCNN [7], Fast RCNN [29] and Faster RCNN [8]. These methods obtain state-of-the-art performances on general object detection and the codes are publicly available.

We adopt the trained car model in voc-release5 [65] for DPM, while train other CNN based models and our method based on the pretrained VGG16 model. All the images containing cars in the trainval and test sets (totally 1434 images) in the PASCAL VOC 2007 dataset are extracted to be evaluated. The car detection evaluation criterion is the same as PASCAL object detection. Intersection over Union (IoU) is set as 0.7 to

[2] http://www.cvlibs.net/datasets/kitti/backups/2017_04_23_01_47_47_object.html

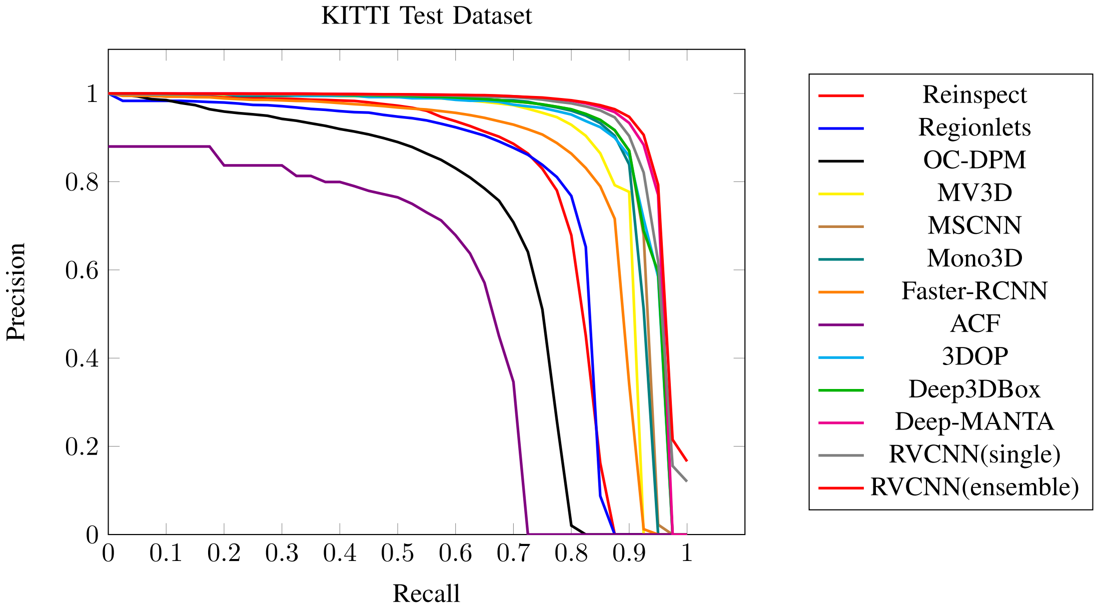


Fig. 7. Precision-recall curves on the KITTI test set of the moderately difficult results. Anonymous submissions without method description are ignored.

TABLE II

COMPARISONS BETWEEN DIFFERENT METHODS ON THE KITTI TEST SET. OUR SINGLE MODEL IS BASED ON GOOGLENET. THE RESULTS OF OUR ENSEMBLE MODEL IS THE DIRECT FUSION OF DETECTION RESULTS GENERATED BY THREE DIFFERENT CNN ARCHITECTURE, GOOGLENET, VGGNET AND RES-50 NET. ANONYMOUS SUBMISSIONS WITHOUT METHOD DESCRIPTION ARE IGNORED

method	Easy	Moderate	Hard
ACF [52]	55.89	54.74	42.98
DPM [5]	68.02	56.48	44.18
DPM-VOC+VP [53]	74.95	64.71	48.76
OC-DPM [54]	74.94	65.95	53.86
SubCat [20]	84.14	75.46	59.71
Regionlets [55]	84.75	76.45	59.70
Reinspect [56]	88.13	76.65	66.23
AOG [57]	84.80	75.94	60.70
spLBP [58]	87.18	77.39	60.59
3DVP [13]	87.46	75.77	65.38
MV3D [59]	89.11	87.67	79.54
RefineNet [60]	89.88	79.17	66.38
Faster R-CNN [8]	86.71	81.84	71.12
3DOP [61]	93.04	88.64	79.10
Mono3D [62]	92.33	88.66	78.96
SDP+RPN [63]	90.14	88.85	78.38
MS-CNN [15]	90.03	89.02	76.11
SubCNN [46]	90.81	89.04	79.27
Deep3DBox [17]	92.98	89.04	77.17
Deep MANTA [16]	**95.77**	90.03	80.62
RRC(ensemble) [64]	93.66	90.19	**86.91**
Our single model	90.82	90.70	84.27
Our ensemble model	91.28	**91.67**	85.43

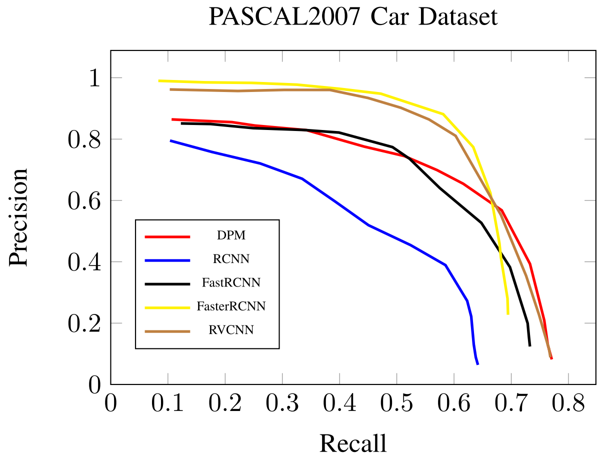


Fig. 8. Precision-recall curves on the PASCAL2007 car dataset.

assess a correct localization. Fig 8 presents precision-recall curves on the PASCAL VOC2007 car testset. Since the 3DVP needs ground truth 3D annotations (cuboids) and camera parameters and we didn't find these labels for PASCAL VOC. Therefore, we removed the subcategory classification task in the experiments on PASCAl VOC dataset. The APs are 63.91% (our model), 38.52% (RCNN), 52.95%(Fast RCNN), 59.82% (Faster RCNN) and 57.14% (DPM) respectively. Though the dataset is very small, our method still outperforms other methods.

V. CONCLUSION

In this paper, we developed a novel vehicle detection scheme based on multi-task deep convolutional neural networks (CNN) and region-of-interest (RoI) voting. Experimental results on the real-world computer vision benchmark KITTI and the PASCAL2007 car dataset show that our method outperforms most existing frameworks for vehicle detection. In the future, we will explore an end-to-end framework for more effective voting mechanism. In addition, we want to combine the CNN model with some occlusion reasoning methods to handle those difficult cases better.

REFERENCES

[1] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Sep. 2009.

[2] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The Kitti vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 3354–3361.

[3] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vis.*, vol. 57, no. 2, pp. 137–154, 2004.

[4] V. Vilaplana, F. Marques, and P. Salembier, "Binary partition trees for object detection," *IEEE Trans. Image Process.*, vol. 17, no. 11, pp. 2201–2216, Nov. 2008.

[5] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.

[6] Y. Li, S. Wang, Q. Tian, and X. Ding, "Learning cascaded shared-boost classifiers for part-based object detection," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1858–1871, Apr. 2014.

[7] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[9] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *Int. J. Comput. Vis.*, vol. 104, no. 2, pp. 154–171, Apr. 2013.

[10] P. Krähenbühl and V. Koltun, "Geodesic object proposals," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 725–739.

[11] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 391–405.

[12] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, "Facial landmark detection by deep multi-task learning," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 94–108.

[13] Y. Xiang, W. Choi, Y. Lin, and S. Savarese, "Data-driven 3D Voxel patterns for object category recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1903–1911.

[14] S. Liu, C. Lu, and J. Jia, "Box aggregation for proposal decimation: Last mile of object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 2569–2577.

[15] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 354–370.

[16] F. Chabot, M. Chaouch, J. Rabarisoa, C. Teulière, and T. Chateau, "Deep manta: A coarse-to-fine many-task network for joint 2D and 3D vehicle analysis from monocular image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 2040–2049.

[17] A. Mousavian, D. Anguelov, J. Flynn, and J. Kosecka, "3D bounding box estimation using deep learning and geometry," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 7074–7082.

[18] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 5325–5334.

[19] Z. Cai, M. Saberian, and N. Vasconcelos, "Learning complexity-aware cascades for deep pedestrian detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 3361–3369.

[20] E. Ohn-Bar and M. M. Trivedi, "Learning to detect vehicles by clustering appearance patterns," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2511–2521, Oct. 2015.

[21] S. Sivaraman and M. M. Trivedi, "Vehicle detection by independent parts for urban driver assistance," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1597–1608, Dec. 2013.

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[23] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Dec. 2014, pp. 1725–1732.

[24] Z. Cui, H. Chang, S. Shan, B. Zhong, and X. Chen, "Deep network cascade for image super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 49–64.

[25] L. Xu, J. S. Ren, C. Liu, and J. Jia, "Deep convolutional neural network for image deconvolution," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1790–1798.

[26] X. Guo, S. Singh, H. Lee, R. L. Lewis, and X. Wang, "Deep learning for real-time atari game play using offline Monte-Carlo tree search planning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3338–3346.

[27] C. Szegedy, A. Toshev, and D. Erhan, "Deep neural networks for object detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 2553–2561.

[28] 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.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.

[29] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1440–1448.

[30] S. Gidaris and N. Komodakis, "LocNet: Improving localization accuracy for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Dec. 2016, pp. 789–798.

[31] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 779–788.

[32] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 21–37.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[34] W. Chu and D. Cai. (Apr. 2016). "Deep feature based contextual model for object detection." [Online]. Available: https://arxiv.org/abs/1604.04048

[35] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 379–387.

[36] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 7263–7271.

[37] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, "TextBoxes: A fast text detector with a single deep neural network," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 4161–4167.

[38] H. T. Niknejad, T. Kawano, Y. Oishi, and S. Mita, "Occlusion handling using discriminative model of trained part templates and conditional random field," in *Proc. IEEE Intell. Veh. Symp. (IV)*, Jun. 2013, pp. 750–755.

[39] B. Li, W. Hu, T. Wu, and S.-C. Zhu, "Modeling occlusion by discriminative and-or structures," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2560–2567.

[40] T. Wu, B. Li, and S.-C. Zhu, "Learning and-or model to represent context and occlusion for car detection and viewpoint estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 9, pp. 1829–1843, Sep. 2016.

[41] Y. Zhou, L. Liu, L. Shao, and M. Mellor, "DAVE: A unified framework for fast vehicle detection and annotation," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 278–293.

[42] C.-H. Kuo and R. Nevatia, "Robust multi-view car detection using unsupervised sub-categorization," in *Proc. Workshop Appl. Comput. Vis. (WACV)*, Dec. 2009, pp. 1–8.

[43] C. Gu and X. Ren, "Discriminative mixture-of-templates for viewpoint classification," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 408–421.

[44] T. Lan, M. Raptis, L. Sigal, and G. Mori, "From subcategories to visual composites: A multi-level framework for object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 369–376.

[45] M. Hoai and A. Zisserman, "Discriminative sub-categorization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1666–1673.

[46] Y. Xiang, W. Choi, Y. Lin, and S. Savarese, "Subcategory-aware convolutional neural networks for object proposals and detection," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2017, pp. 924–933.

[47] X. Li *et al.*, "DeepSaliency: Multi-task deep neural network model for salient object detection," *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3919–3930, Aug. 2016.

[48] Y. Cao, C. Shen, and H. Shen, "Exploiting depth from single monocular images for object detection and semantic segmentation," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 836–846, Feb. 2017.

[49] D. Yoo, S. Park, J.-Y. Lee, A. S. Paek, and I. So Kweon, "AttentionNet: Aggregating weak directions for accurate object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 2659–2667.

[50] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. Multimedia*, 2014, pp. 675–678.

[51] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.

[52] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1532–1545, Aug. 2014.

[53] B. Pepik, M. Stark, P. Gehler, and B. Schiele, "Multi-view and 3D deformable part models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 11, pp. 2232–2245, Nov. 2015.

[54] B. Pepikj, M. Stark, P. Gehler, and B. Schiele, "Occlusion patterns for object class detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3286–3293.

[55] X. Wang, M. Yang, S. Zhu, and Y. Lin, "Regionlets for generic object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 17–24.

[56] R. Stewart, M. Andriluka, and A. Y. Ng, "End-to-end people detection in crowded scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2325–2333.

[57] B. Li, T. Wu, and S.-C. Zhu, "Integrating context and occlusion for car detection by hierarchical and-or model," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 652–667.

[58] Q. Hu, S. Paisitkriangkrai, C. Shen, A. van den Hengel, and F. Porikli, "Fast detection of multiple objects in traffic scenes with a common detection framework," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 4, pp. 1002–1014, Apr. 2016.

[59] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3D object detection network for autonomous driving," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 1907–1915.

[60] R. N. Rajaram, E. Ohn-Bar, and M. M. Trivedi, "RefineNet: Iterative refinement for accurate object localization," in *Proc. Intell. Transp. Syst. Conf.*, Nov. 2016, pp. 1528–1533.

[61] X. Chen *et al.*, "3D object proposals for accurate object class detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 424–432.

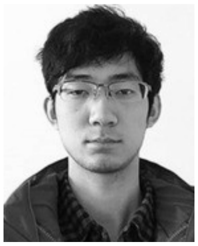
[62] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun, "Monocular 3D object detection for autonomous driving," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2147–2156.

[63] F. Yang, W. Choi, and Y. Lin, "Exploit all the layers: Fast and accurate CNN object detector with scale dependent pooling and cascaded rejection classifiers," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2129–2137.

[64] J. Ren *et al.*, "Accurate single stage detector using recurrent rolling convolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 5420–5428.

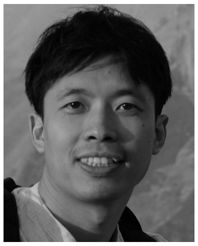
[65] R. B. Girshick, P. F. Felzenszwalb, and D. McAllester, *Discriminatively Trained Deformable Part Models, Release 5*, 2012. [Online]. Available: http://people.cs.uchicago.edu/~rbg/latent-release5/

Wenqing Chu received the B.E. degree in computer science and technology from the Huazhong University of Science and Technology in 2014. He is currently pursuing the Ph.D. degree with the State Key Laboratory of CAD&CG, College of Computer Science, Zhejiang University, China. His research interests include machine learning, computer vision, and data mining.



Yao Liu received the B.E. degree from Southeast University, Nanjing, China, in 2014. He is currently pursuing the master's degree with the State Key Laboratory of CAD&CG, Zhejiang University. His current research interests include computer vision and machine learning.

Chen Shen received the B.S. degree in electrical engineering from Zhejiang University, China, in 2012. He is currently pursuing the Ph.D. degree with the School of Instrumental Engineering, Zhejiang University. Since 2016, he has been with Alibaba Inc., Hangzhou, China, as an Intern. His research interests include deep learning, machine learning, and computer vision.



Deng Cai received the Ph.D. degree in computer science from the University of Illinois at Urbana–Champaign in 2009. He is currently a Professor with the State Key Laboratory of CAD&CG, College of Computer Science, Zhejiang University, China. His research interests include machine learning, data mining, and information retrieval.

Xian-Sheng Hua (F'16) received the B.S. and Ph.D. degrees in applied mathematics from Peking University, Beijing, in 1996 and 2001, respectively. In 2001, he joined Microsoft Research Asia as a Researcher. He has been a Principal Research and Development Lead in multimedia search for the Microsoft search engine, Bing, since 2011, where he led a team that designed and delivered leading-edge media understanding, indexing, and searching features. He has been a Senior Researcher of Microsoft Research Redmond since 2013, where he was involved in Web-scale image and video understanding and search, and related applications. He became a Researcher and the Senior Director of the Alibaba Group in 2015, leading the Multimedia Technology Team in the Search Division. He has authored or co-authored over 250 research papers in these areas and has filed over 90 patents. His research interests have been in the areas of multimedia search, advertising, understanding, and mining, and pattern recognition and machine learning. He was honored as one of the recipients of the prestigious 2008 MIT Technology Review TR35 Young Innovator Award for his outstanding contributions to video search. He received the Best Paper and Best Demonstration Awards at ACM Multimedia 2007, the Best Student Paper Award at ACM CIKM 2009, the Best Paper Award at MMM 2010, the Best Demonstration Award at ICME 2014, and the Best Paper Award of the IEEE TRANSACTIONS ON CSVT in 2014. He served as a Program Co-Chair for the IEEE ICME 2013, the ACM Multimedia 2012, and the IEEE ICME 2012, and on the Technical Directions Board of the IEEE Signal Processing Society. He is an ACM Distinguished Scientist. He served or is now serving as an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA and the *ACM Transactions on Intelligent Systems and Technology*.