

DOI:10.3969/j.issn.1001-4551.2018.08.019

基于 RCNN 的车辆检测方法研究*

朱茂桃¹, 张鸿翔^{1,2}, 方瑞华²

(1. 江苏大学 汽车与交通工程学院, 江苏 镇江 212013; 2. 上海干巷车镜实业有限公司, 上海 201518)

摘要:为了解决基于传统机器学习车辆检测算法实时性和泛化能力差的问题,对基于深度学习的车辆检测算法进行研究。分析了 Faster R-CNN 检测算法原理,使用 Python 编程语言基于深度学习框架 TensorFlow 实现了 Faster R-CNN 算法;采集了4个季节不同天气情况下的12 000张路况图片数据,并对数据集进行了标注,采用3种不同方式对数据集进行了预处理;通过对照试验对 Faster R-CNN 算法中的超参数进行了调优;使用控制变量法比较了 RCNN、SPPnet、Fast R-CNN 和 Faster R-CNN 4种算法的检测准确率和检测速度,指出了4种算法的主要耗时步骤,验证了 Faster R-CNN 车辆检测算法的有效性。研究表明:基于 Faster R-CNN 的车辆检测算法达到每张69 ms的检测速度和91.3%的准确率,能够实现实时高精度的车辆检测。

关键词:汽车工程;辅助驾驶;车辆检测;深度学习;区域提议网络;卷积神经网络

中图分类号:TP301.6;U 471.15

文献标志码:A

文章编号:1001-4551(2018)08-0880-06

Algorithm of vehicle detection based on RCNN

ZHU Mao-tao¹, ZHANG Hong-xiang^{1,2}, FANG Rui-hua²

(1. School of Automobile and Traffic Engineering, Jiangsu University, Zhenjiang 212013, China;

2. Shanghai Ganxiang Automobile Mirror Industry, Shanghai 201518, China)

Abstract: Aiming at improving the robust adaptation and real-time of the vehicle detection algorithm based on traditional machine learning, vehicle detection algorithm based on deep learning was researched. The principle of Faster R-CNN detection algorithm was analyzed. Based on TensorFlow deep learning frame and made use of Python programming language, the Faster R-CNN algorithm was realized. The road condition data set of four seasons was collected and labeled, 12 000 pictures was included. The data set was pretreat by three different method and the algorithm parameter of Faster R-CNN was tuned by controlled experiment. The accuracy and speed of four detection algorithm, RCNN, SPPnet, Fast R-CNN and Faster R-CNN, was compared by variable-controlling approach. The results indicate that the speed of vehicle detection based on Faster R-CNN is 69 ms and the accuracy rate is 91.3%, the algorithm can realize real-time and high accuracy of vehicle detection.

Key words: vehicle engineering; assistance driving; vehicle detection; deep learning; region proposal networks; convolutional neural networks

0 引 言

近年来,随着人们对车辆安全性能的关注,高级辅助驾驶系统(ADAS)受到车企的广泛关注。因此基于计算机视觉的车辆检测算法具有重要价值。

传统车辆检测方法的步骤为:在给定的图像上选

取感兴趣区域,根据特征算子(常用的有 SIFT、HOG 等)对感兴趣区域提取特征^[1-2],使用训练的分类器进行分类^[3-4]。MU K 等^[5]针对基于边缘的车辆检测方法容易受到噪声和背景的干扰,导致检测不准确的问题,提出了一种基于多尺度边缘融合的车辆检测方法;WU B F 等^[6]提出了一种实时的车道线检测算法和基于模糊逻辑的车辆检测算;KIM Z W^[7]提出了基于概

收稿日期:2017-12-01

基金项目:上海科技人才计划项目(16XD1420900)

作者简介:朱茂桃(1963-),男,江苏镇江人,博士,教授,主要从事车身工程、智能驾驶辅助方面的研究。E-mail:13775546980@139.com

率特征分组的实时车辆检测方法并将其运用于车辆跟踪。但是传统目标检测存在两个主要问题:(1)基于滑动窗口的区域选择策略缺乏针对性,耗时长;(2)人工提取的特征没有很好的泛化能力。

2012 年开始,各种视觉任务都启用了卷积神经网络(CNN)来构建新的模型。CNN 优于传统算法的 3 个主要因素:(1)大规模的训练数据;(2)网络结构更加复杂、更深、参数更多;(3)GPU 对计算的加速,使以往需要数周的训练过程只需一天甚至数小时以内就可以完成。ADAS 的视觉感知部分属于计算机视觉的范畴,因此,很多传统的车辆检测算法正在被深度学习所取代。

本文根据 RCNN 系列深度学习检测算法建立数据集,对 Faster R-CNN 算法调参,最后比较 RCNN、SPPnet、Fast R-CNN 和 Faster R-CNN 算法的检测效果^[8-11]。

1 RCNN 系列深度学习检测算法

针对卷积神经网络的问题,GIRSHICK R 提出了 RCNN 采用 Selective Search 方法,得到可能存在检测目标的若干图像局部区域^[12],然后将这些区域分别输入到 CNN 中得到区域特征,在特征图上接分类器,判断对应的区域是否属于要检测的目标,最后针对标定框进行回归以修正预测框的位置。但 RCNN 存在着重复计算的问题。HE K 将空间金字塔池化层引入 CNN 提出了 SPPnet,降低了 CNN 网络对输入图片尺寸的限制,提高了准确率。基于 SPPnet 的思路,GIRSHICK R 又提出了 Fast-RCNN,采用自适应尺度池化,

将候选区域映射到 CNN 最后一个卷积层的特征图上,只需要提取一次特征,从而提高了检测速度。但 Fast R-CNN 存在提取候选区域耗时长的问题,因此 GIRSHICK R 等人提出了 Faster R-CNN 算法。

2 基于 Faster R-CNN 的车辆检测算法

Faster R-CNN 算法由两大模块组成:(1)RPN 候选框提取模块;(2)Fast R-CNN 检测模块。其中,RPN 用于提取候选框;Fast R-CNN 基于 RPN 提取的 proposal 检测并识别 proposal 中的目标。

与 SPPNet 和 Fast R-CNN 相比,Faster R-CNN 采用 CNN 提取特征和候选区域,并将两个网络权值共享、联合调优,实现了端到端的目标检测,既解决了提取候选区域耗时长的问题,又保证了检测的准确率。

2.1 模型训练

2.1.1 CNN 网络架构选取

CNN 网络有两个可选方案:(1)选择经典的 VGG16^[13];(2)选择 ZFnet^[14]。经过测试 ZFnet 精度为 58.5%,VGG16 精度为 66%。VGG 网络的精度高,但计算量大。为了减少计算量,选用 ZFnet。ZFnet 是基于 Alexnet 的微调版本^[15],其 top-5 错误率为 11.2%。ZFnet 使用 ReLu 作为激活函数,使用了交叉熵代价函数,通过使用较小的过滤器以保留更多的原始像素信息。

网络结构如图 1 所示。

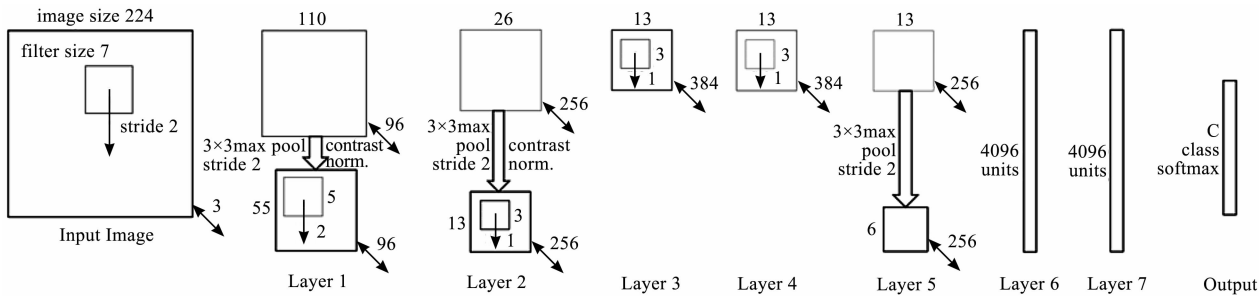


图 1 ZFnet 网络结构

ZFnet 特征提取部分包含了 5 个卷积层、2 个全连接层,在 ZFnet 中第 5 个池化层包含 256 个通道,称为特征图,共有神经元个数为 9 216 (256 × 6 × 6),第 6 和第 7 全连接层的神经元个数都是 4 096,通过这个网络训练完毕后,最后提取特征。

2.1.2 训练 RPN 网络

RPN 原理如图 2 所示。

实现方式:在最后一层卷积的特征图上用一个 3 × 3 的滑窗生成一个长度为 256 (对应于 ZFnet) 维长度

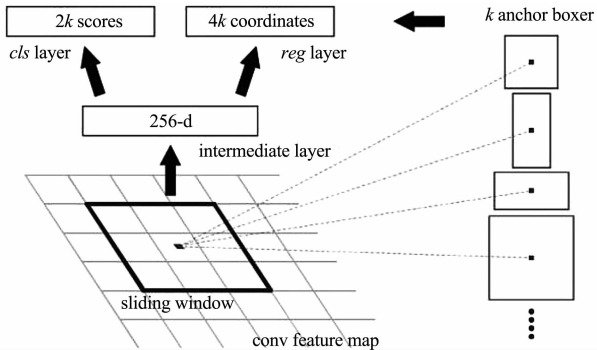


图 2 RPN 原理图

的全连接特征,然后在这个 256 维的特征后产生两个分支的全连接层:(1) 分类层用于判定该候选区域是前景还是背景;(2) 回归层用于预测候选区域的中心锚点对应的预测框坐标 x, y 和宽高 w, h 。采用滑动窗口的方式保证了分类层和回归层关联了最后一次卷积的全部特征图。然后用分别为 $2 \times k$ 和 $4 \times k$ 个大小为 1×1 步长为 1 的卷积核实现上一层特征到分类层和回归层的特征映射。

图中的 anchor 字面上可以理解为锚点,位于之前提到的 $n \times n$ 的滑窗中心处。对于每个滑窗可以同时预测 k 个区域。 k 个区域即 k 个预测窗口,每一个预测窗口又可以用一个比例,一个高宽比和滑窗中的锚点唯一确定。用 3 种比例和 3 种高宽比确定出当前锚点位置处对应的 9 个预测窗口,即 $k = 9$ 。

Anchor 正负样本标定规则如下:

① 对每个标定框,与其重叠比例最大的 anchor 记为正样本(保证每个标定框至少对应一个正样本 anchor);

② 对 ① 剩余的 anchor,如果其与某个标定区域重叠比例大于 0.7,记为正样本(每个标定框可能会对应多个正样本 anchor,但每个正样本 anchor 只可能对应一个标定框);如果其与任意一个标定框的重叠比例都小于 0.3,记为负样本;

③ 对 ①、② 剩余的 anchor,弃去不用;

④ 跨越图像边界的 anchor,弃去不用。

2.1.3 定义损失函数

分类损失:每个 anchor 后面接有一个二分类 softmax,得到 anchors 的分类概率。其公式如下:

$$L(\{p_i\}) = \frac{1}{N_{\text{cls}}} \sum_i L_{\text{cls}}(p_i, p_i^*) \quad (1)$$

式中: i —第 i 个 anchor; p_i —分类概率; L_{cls} —分类损失; N_{cls} —分类样本数,当 anchor 为正样本时 $p_i^* = 1$,负样本则为 $p_i^* = 0$ 。

回归损失:在每个 anchor 后接上一个回归函数,计算预测窗口与标定窗口之间的损失。其公式如下:

$$L(\{t_i\}) = \frac{1}{N_{\text{reg}}} \sum_i p_i^* L_{\text{reg}}(t_i, t_i^*) \quad (2)$$

式中: t_i —预测窗口的尺寸; t_i^* —标定窗口的尺寸; L_{reg} —回归损失; N_{reg} —回归样本数。

计算出 t_i 和 t_i^* :

(1) 预测框,即 RPN 网络预测出区域的中心位置坐标和宽高;

(2) 锚点预测窗口:9 个锚点对应 9 个不同比例和高宽比的预测窗口,每个预测窗口都有一个中心点位

置坐标和宽高;

(3) 目标位置:标定框也对应一个中心点位置坐标和宽高。

$$t_x = \frac{(x - x_a)}{w_a} \quad t_y = \frac{(y - y_a)}{h_a}$$

$$t_w = \log\left(\frac{w}{w_a}\right) \quad t_h = \log\left(\frac{h}{h_a}\right) \quad (4)$$

$$t_x^* = \frac{(x^* - x_a)}{w_a} \quad t_y^* = \frac{(y^* - y_a)}{h_a}$$

$$t_w^* = \log\left(\frac{w^*}{w_a}\right) \quad t_h^* = \log\left(\frac{h^*}{h_a}\right) \quad (5)$$

式中: (x, y) —预测框的中心坐标; (w, h) —预测框的宽高; (x_a, y_a) —锚点预测窗口的中心坐标; (w_a, h_a) —锚点预测窗口的宽高; (x^*, y^*) —标定框的中心坐标; (w^*, h^*) —标定框的宽高。

定义损失函数:RPN 的损失函数是由分类损失和回归损失按一定比重组成。RPN 总损失函数为:

$$L(\{p_i\} \{t_i\}) = \frac{1}{N_{\text{cls}}} \sum_i L_{\text{cls}}(p_i, p_i^*) + \lambda \frac{1}{N_{\text{reg}}} \sum_i p_i^* L_{\text{reg}}(t_i, t_i^*) \quad (6)$$

式中: λ —比例系数。

2.1.4 RPN 和 CNN 参数共享和联合调优

因为分别训练 RPN 和 Fast R-CNN 两种不同任务的网络模型,会导致无法共享网络权重,所以需要采用交替训练方法:

(1) 用 ImageNet 模型初始化,独立训练一个 RPN 网络;

(2) 仍然用 ImageNet 模型初始化,但是使用上一步 RPN 网络产生的区域作为输入,训练一个 Fast R-CNN 网络,至此,两个网络每一层的参数完全不共享;

(3) 使用第二步的 Fast R-CNN 网络参数初始化一个新的 RPN 网络,但是把 RPN、Fast-RCNN 共享的那些卷积层的学习率设置为 0,也就是不更新,仅仅更新 RPN 特有的那些网络层,重新训练,此时,两个网络已经共享了所有公共的卷积层;

(4) 仍然固定共享的那些网络层,把 Fast R-CNN 特有的网络层也加入进来,形成一个联合网络,继续训练,微调 Fast R-CNN 特有的网络层,此时,该网络已经实现了内部预测区域并实现检测的功能。

2.2 车辆检测

Faster R-CNN 运用同一个 ZFnet 网络将分离的 RPN 定位和 CNN 分类进行结合,从而实现了端到端的目标检测。实现步骤为:

- (1)给定输入图像,经过卷积操作得到最后一层的卷积特征图;
- (2)使用滑动窗口对特征图进行卷积,在最后一层卷积层上得到候选区域;
- (3)对(2)得到的候选区域进行非最大值抑制(NMS^[16]),得到评分前 300 的候选区域;
- (4)在提取的特征向量后面接分类层和回归层分别计算分类评分和边框回归评分。

3 实验及结果分析

本文采用 Faster R-CNN 算法对车辆进行检测。

3.1 训练样本处理

数据集由采集的路况图片数据组成,包含了 4 个季节各种天气状况下的图片。原始影像大小为 1 280 像素 × 720 像素。

3.2 数据集的构建

本试验的训练集由行车过程中采集的车辆正样本与负样本各 4 000 张,共 8 000 张组成;验证集由正负样本各 1 000 张,共 2 000 张组成;测试集由正负样本各 1 000 张,共 2 000 张组成。其中验证集是用于调参,测试集用于检验算法的好坏。

使用 LabelImg 分别对训练集、验证集和测试集图片上的车辆进行统一的标注,标注界面如图 3 所示。



图 3 图片标定示例

为了改善图像数据,进行数据增强。对图像进行 3 种不同的随机变换:随机裁剪、随机翻转、随机颜色失真。

本次试验中,采用查准率、查全率计算出平均准确率来进行评价:

$$P_{re} = \frac{T_p}{T_p + F_p}$$
(7)

$$R_{ec} = \frac{T_p}{T_p + F_N}$$
(8)

$$mAP = \int_0^1 P_{re}(R_{ec}) d(R_{ec})$$
(9)

式中: T_p — 正确的正例; F_p — 错误的正例; F_N — 错误的反例; P_{re} — 查准率; R_{ec} — 查全率; mAP — 平均准确率。

在测试阶段,当预测框与标定框重叠面积达到标记外围框的 90% 以上时,视为检测成功。

3.3 实验结果

本文使用 Tensorflow 深度学习框架实现 Faster R-CNN 算法。首先使用 3.2 节创建的数据训练集和验证集对模型进行调参,然后通过控制变量法对 Faster R-CNN 的参数进行优化,研究其对 mAP 的影响,最后使用测试集对 RCNN、SPPnet、Faster R-CNN 与 Faster R-CNN 算法进行测试和对比。实验用计算机 CPU 为 Intel E5-2620 V3, GPU 为 GeFore GTX 1080ti,软件环境为 Python3.6。

Faster-RCNN 算法需要设置一些超参数来对 mAP 进行调优:dropout 值、批处理(batch size)、NMS 前后候选区域个数。试验结果如表(1~3)所示。

Dropout 在模型训练时随机让网络中某些隐含层节点的权重不产生作用,是用来防止过拟合的一个参数。Dropout 值对 mAP 的影响与数据集大小相关,调试 Dropout 值使 mAP 最大。表 1 说明:在将 Dropout 的值从 20% 提高到 80% 的过程中,当 Dropout 值取 60% 时,mAP 取最大值 0.835。

表 2 说明:当候选区域数量减少时,mAP 也逐渐减少。这是因为 NMS 是用来选取候选区域里分数最高,

表 1 Dropout 值对 mAP 影响

Droupout 值/(%)	区域提议阶段批尺寸	检测阶段批尺寸	NMS 前候选区域个数	NMS 后候选区域个数	mAP
20	200	100	2 000	250	0.831
30	200	100	2 000	250	0.815
40	200	100	2 000	250	0.821
50	200	100	2 000	250	0.793
60	200	100	2 000	250	0.835
70	200	100	2 000	250	0.786

表 2 不同 NMS 数目对 mAP 影响

Droupout 值/(%)	区域提议阶段批尺寸	检测阶段批尺寸	NMS 前候选区域个数	NMS 后候选区域个数	mAP
60	200	100	2 000	300	0.825
60	200	100	1 500	250	0.817
60	200	100	1 000	200	0.809
60	200	100	500	150	0.801

表 3 不同的 batch size 对 mAP 影响

Droupout 值/(%)	区域提议阶段批尺寸	检测阶段批尺寸	NMS 前候选区域个数	NMS 后候选区域个数	mAP
60	200	100	2 000	300	0.839
60	100	50	20 00	300	0.834
60	50	25	2 000	300	0.829
60	25	12	2 000	300	0.825

并且抑制那些分数低的窗口。如果候选区域数目减少,经 NMS 后,由于取样少,将导致预测框不能很好地框出图中车辆位置,从而导致 *mAP* 下降。但如果候选

区域数量过多,会导致计算量很大,故一般取 2 000 个 NMS 前候选区域(300 个 NMS 后候选区域个数)为宜。NMS 前后预测框的对比如图 4 所示。

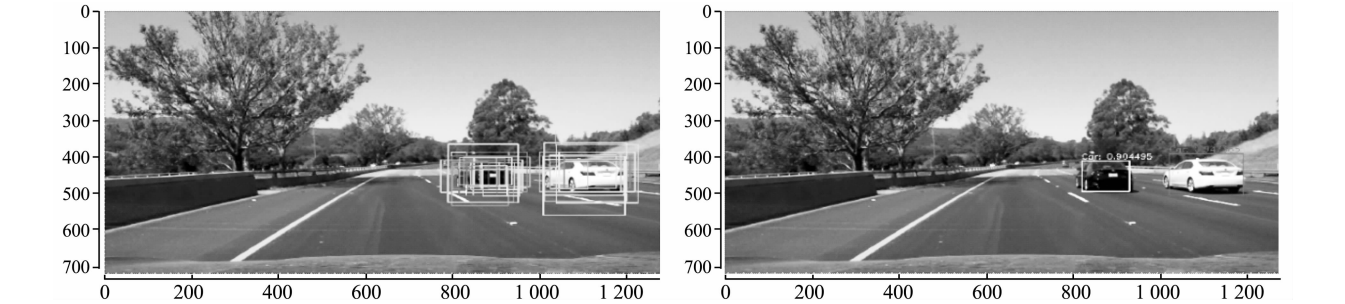


图 4 NMS 前后预测框对比图

保持 Dropout 值和候选区域个数不变,研究 batch size 与 *mAP* 之间的关系。表 3 说明:当 batch size 减少时,*mAP* 也逐渐减小。这是因为采用 batch 是梯度下降算法和随机梯度下降的折中方案。batch size 越大越接近梯度下降算法,*mAP* 越高,但计算耗时长,极端的,当 batch 为整个训练集时,采用 batch 等价于梯度下降算法;batch size 越小越接近随机梯度下降,*mAP* 越低,但计算速度快,当 batch size 为 1 时,采用 batch 等价于随机梯度下降算法^[17]。综合考虑这两方面因素,取区域提议阶段 batch size 为 100,检测阶段 batch size 为 50。

分别使用 RCNN、SPPnet、Fast R-CNN 和 Faster R-CNN 等 4 种算法对车辆进行检测,4 种算法在测试集上的正确率和召回率如图 5 所示。可以看出:基于 RCNN 检测算法的正确率得到提高。从 RCNN 到 SPPnet 解决了不同尺寸输入图片转化成相同尺度向量的问题,从 SPPnet 到 Fast R-CNN 节省了对整张图片进行 CNN 特征提取的操作,从 Fast R-CNN 到 Faster R-CNN 提出了 RPN,并将 RPN 和 Fast R-CNN 共享网络,进一步提高了准确率。

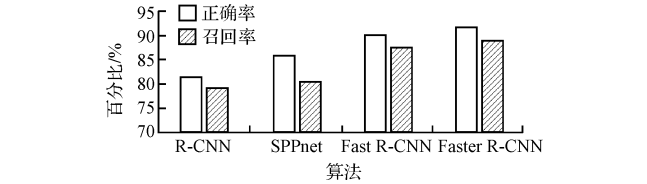


图 5 4 种算法的检测精度对比图

采用同一台服务器对本文所有试验测试,测试集图片大小为 1 280 像素×720 像素。表 4 说明 RCNN、SPPnet 和 Fast R-CNN 都采用 Selective Search 方法提取候选区域耗时长;Faster R-CNN 在 Fast R-CNN 基础上采用卷积特征与 RPN 共享的方法,使得检测时间缩短到了 69 ms 左右。

实验结果表明:基于深度学习的 Faster R-CNN 算法可以满足车辆检测的实时性要求。

表 4 4 种算法的检测速度对比图

算法	候选区域数量	关键步骤	步骤耗时	速度
RCNN	2 000	selective search + 提取特征	36 s/张	47 s
SPPnet	2 000	Selective Search	23.1 s/张	24.5 s
Fast R-CNN	2 000	Selective Search	1.7 s/张	2 s
Faster R-CNN	2 000	提取特征 + region proposal	47 ms/张	69 ms

4 结束语

本研究针对车辆检测对于高实时性和高准确率的要求,分析了 RCNN 系列深度学习算法在车辆检测领域的有效性。结果表明:基于 Faster R-CNN 的车辆检测算法可以达到每张 69 ms 的识别速度和 91.3% 的准确率,提高了实时检测车辆的效果。

参考文献(References):

- [1] MO G, ZHANG Y, ZHANG S, et al. A method of vehicle detection based on sift features and boosting classifier[J]. **Journal of Convergence Information Technology**, 2012, 7(12):328-334.
- [2] XU Y, YU G, WANG Y, et al. A hybrid vehicle detection method based on Viola-Jones and HOG + SVM from UAV images[J]. **Sensors**, 2016, 16(8):1325.
- [3] ANTONY C, KONGUVEL E. Vehicle detection system using svm classification and haar filter[J]. **Revista Espanola De Pedagogia**, 2014, 73(260):157-175.
- [4] BAEK Y M, KIM W Y. Forward vehicle detection using cluster-based AdaBoost[J]. **Optical Engineering**, 2014, 53(10):102103.
- [5] MU K, HUI F, ZHAO X, et al. Multiscale edge fusion for vehicle detection based on difference of Gaussian[J]. **Optik - International Journal for Light and Electron Optics**, 2016, 127(11):4794-4798.
- [6] WU B F, LIN C T, CHEN C J. Real-time lane and vehicle detection based on a single camera model[J]. **International Journal of Computers & Applications**, 2015, 32(2):149-159.
- [7] KIM Z W, MALIK J. Fast vehicle detection with probabilistic feature grouping and its application to vehicle tracking[C]. **IEEE International Conference on Computer Vision**, Nice: IEEE, 2003.
- [8] GIRSHICK R, DONAHUE J, DARRELL T, et al. Region-

based convolutional networks for accurate object detection and segmentation[J]. **IEEE Transactions on Pattern Analysis & Machine Intelligence**, 2016, 38(1):142-158.

- [9] HE K, ZHANG X, REN S, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition [J]. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, 2015, 37(9):1904-1916.
- [10] GIRSHICK R. Fast R-CNN[C]. **2015 IEEE International Conference on Computer Vision**, Santiago:IEEE, 2015.
- [11] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[J]. **IEEE Transactions on Pattern Analysis & Machine Intelligence**, 2017, 39(6):91-99.
- [12] SANDE K E A V D, UIJLINGS J R R, GEVERS T, et al. Segmentation as selective search for object recognition[J]. **IEEE International Conference on Computer Vision**, 2012, 104(2):1879-1886
- [13] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition[D]. Oxford:Department of Engineering Science, University of Oxford, 2014.
- [14] ZEILER M D, FERGUS R. Visualizing and understanding convolutional networks [D]. New York:Department of Computer Science, New York University, 2014.
- [15] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Image Net classification with deep convolutional neural networks [J]. **Communications of the ACM**, 2012, 60(2):1097-1105.
- [16] NEUBECK A, GOOL L V. Efficient non-maximum suppression[C]. **18th International Conference on Pattern Recognition**, Hong Kong:IEEE, 2006.
- [17] BOTTOU L. Large-scale machine learning with stochastic gradient descent[C]. **Proceeding of COMPSTATS 2010**, 19th International Conference an Computational Statisthis, Paris:Springer, 2010.

[编辑:张豪]

本文引用格式:

朱茂桃,张鸿翔,方瑞华. 基于 RCNN 的车辆检测方法研究[J]. 机电工程, 2018, 35(8):◆-◆,◆.

ZHU Mao-tao, ZHANG Hong-xiang, FANG Rui-hua. Algorithm of vehicle detection based on RCNN[J]. **Journal of Mechanical & Electrical Engineering**, 2018, 35(8):◆-◆,◆.

《机电工程》杂志: <http://www.meem.com.cn>