IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

Vehicle Detection and classification on road

Department of Artificial Intelligence and Data Science, Anantrao Pawar College of Engineering and Research, Pune, Maharashtra, India.

Project Guide Name: Prof Swati.S.Joshi

Abstract:

In this paper, we present an efficient and effective framework for vehicle detection and classification from traffic surveillance cameras. First, we cluster the vehicle scales and aspect ratio in the vehicle datasets. Then, we use convolution neural network (CNN) to detect a vehicle. We utilize feature fusion techniques to concatenate highlevel features and low-level features and detect different sizes of vehicles on different features. In order to improve speed, we naturally adopt fully convolution architecture instead of fully connection (FC) layers. Furthermore, recent complementary advances such as batch-norm, hard example mining, and inception have been adopted. Extensive experiments on JiangSuHighway Dataset (JSHD) demonstrate the competitive performance of our method. Our framework obtains a significant improvement over the Faster R-CNN by 6.5% mean average precision(mAP). With 1.5G GPU memory at test phase, the speed of the network is 15 FPS, three times faster than the Faster R-CNN.

Introduction:

Vehicle detection is a very important component in traffic surveillance and automatic driving [1]. The traditional vehicle detection algorithms such as Gaussian mixed model (GMM) [2] has achieved promising achievements. But it is not ideal due to illumination changes, background clutter, occlusion, etc. Vehicle detection is still an important challenge in computer vision. With the revival of DNN [3], object detection has achieved significant advances in recent years. Current top deep-network-based object detection frameworks can be divided into two categories: the two-stage approach, including [4–8], and one-stage approach, including [9–11]. In the two-stage approach, a sparse set candidate object boxes is first generated by selective search or region proposal network, and then, they are classified and regressed. In the one-stage approach, the network straightforward generated dense samples over locations, scales, and aspect ratios; at the same time, these samples will be classified and regressed. The main advantage of one-stage is real time; however, its detection accuracy is usually behind the two-stage, and one of the main reasons is class imbalance problem [12]. In the two-stage, Region-based Convolutional Network method (R-CNN) is the pioneer of deep-network-based object detection. R-CNN utilizes selective search to generate

2000 candidate boxes; each candidate box is to be warped into fixed size and as an input image of CNN, so 2000 candidate boxes will be computer 2000 times. It has too low efficiency. In order to reduce computer, Fast R-CNN [5] generates candidate boxes on the last layer feature map and adopts RoI pooling. Under Fast R-CNN pipeline, Faster R-CNN [4] shares full-image convolutional feature with the detection network to enable nearly cost-free region proposals. The aforementioned approaches adopt fully connection layers to classify object. It is time-consuming and space-consuming both in training and inference time. R-FCN [8] uses fully convolution and adding position-sensitive score maps. Nevertheless, R-FCN still needs region proposals generated from region proposal network. The aforementioned methods are general object detection methods. However, vehicle detection is special detection. If we straightforwardly use general object detection algorithms to detect vehicles, the effect is not the best. The main reasons are the following three aspects: (1) Faster R-CNN and Single Shot MultiBox Detector (SSD) using aspect ratios are [0.5, 1, 2], but the aspect ratio range of vehicles is not so big. (2) In Faster R-CNN and SSD extract candidate regions on high-level feature map, the high-level feature has more semantic information, but cannot locate well. (3) Vehicle detection requires high real-time, but Faster R-CNN adopts FC layers. It takes about 0.2 s per image for VGG16 [13] network. Aimed to the general object detection methods, there exist problems. We present an efficient and effective framework for vehicle detection and classification from traffic surveillance cameras. This method fuses the advantages of two-stage approach and one-stage approach. Meanwhile, we use some tricks such as hard example mining [14], data augmentation, and inception [15]. The main contributions of our work are summarized as follows: 1) We use k-means algorithm to cluster the vehicle scales and aspect ratios in the vehicle datasets. This process can improve 1.6% mean average precision (mAP). 2) We detect vehicles on different feature map according to different size vehicles. 3) We fuse the low-level and high-level feature map, so the low-level feature map has more semantic information. Our detector is time and resource efficient. We evaluate our framework on JiangSuHighway Dataset (JSHD) (Fig. 1) and obtain a significant improvement over the state-of-the-art Faster R-CNN by 6.5% mAP. Furthermore, our framework achieves 15 FPS on a NVIDIA TITAN XP, three times faster than the seminal Faster R-CNN

Related work:

In this section, we give a brief introduction of vehicle detection in traffic surveillance cameras. Vision-based vehicle detection algorithms can be divided into three categories: motion-based approaches, hand-crafted feature-based approaches, and CNN-based approaches. Motion-based approaches include frame subtraction, optical flow, and background subtraction. Frame subtraction computers the differences of two or three consecutive frames sequences to detect the motion object. Frame subtraction is characterized by simple calculation and adapting dynamic background, but it is not ideal for motion that is too fast or too slow. Optical flow [16] calculates the motion vector of each pixel and tracks these pixels, but this approach is complex and time-consuming. Background subtraction such as GMM are widely used in vehicle detection by modeling the distribution of the background and foreground [2]. However, these approaches cannot classify and detect still vehicles. Hand-crafted feature-based approaches include Histogram of Oriented Gradients (HOG) [17], SIFT [18], and Harr-like. Before the success of CNN-based approaches, hand-crafted feature approaches such as deformable part-based model (DPM) [19] have achieved the state-

of-art performance. DPM explores improved HOG feature to describe each part of vehicle and followed by classifiers like SVM and Adaboost. However, hand-crafted feature approaches have low feature representation. CNN-based approaches have shown rich representation power and achieved promising results [4–6, 9, 11]. R-CNN uses object proposal generated by selective search [20] to train CNN for detection tasks. Under the R-CNN framework, SPP-Net [7] and Fast R-CNN [5] speed up through generating region proposal on feature map; these approaches only need computer once. Faster R-CNN [4] uses region proposal network instead of selective search, then it can train end to end and the speed and accuracy also improve.

R-FCN [8] tries to reduce the computation time with position-sensitive score maps. Considering the high efficiency, the one-stage approach attracts much more attention recently. YOLO [9] uses a single feed-forward convolutional network to directly predict object classes and locations, which is extremely fast. SSD [11] extracts anchors of different aspect ratios and scales on multiple feature maps. It can obtain competitive detection results and higher speed. For example, the speed of SSD is 58PFS on a NVIDIA TITAN X for 300 × 300 input, nine times faster than Faster R-CNN

Methods:

This section describes our object detection framework (Fig. 2). We first introduce k-means algorithm to prepare data in Section 3.1. Then, in Section 3.2, we present feature concatenate to fuse high-level and low-level feature map. Next, we explain how to generate candidate anchor boxes on different feature map in Section 3.3. In Section 3.4, we discuss how to detect different size vehicles on different feature map. Finally, we introduce batch-norm, hard example, and inception; these tricks can improve the result

This section describes our object detection framework (Fig. 2). We first introduce k-means algorithm to prepare data in Section 3.1. Then, in Section 3.2, we present feature concatenate

Bounding box clustering:

The traditional object detection algorithms use sliding window to generate candidate proposal, but these methods are time-consuming. In CNN-based detectors such as Faster R-CNN and SSD use aspect ratios [0.5, 1, 2], so the candidate proposals are less than sliding window. But there are two issues in this way. The first issue is that the aspect ratios are hand-picked. If we pick better priors of dataset, it will be easier for the network to predict good detections. The second issue is that the aspect ratios are designed for general object detection such as PASCAL VOC [21] and COCO [22] dataset. It is not very suitable for vehicle detection. In order to solve these issues, we run k-means

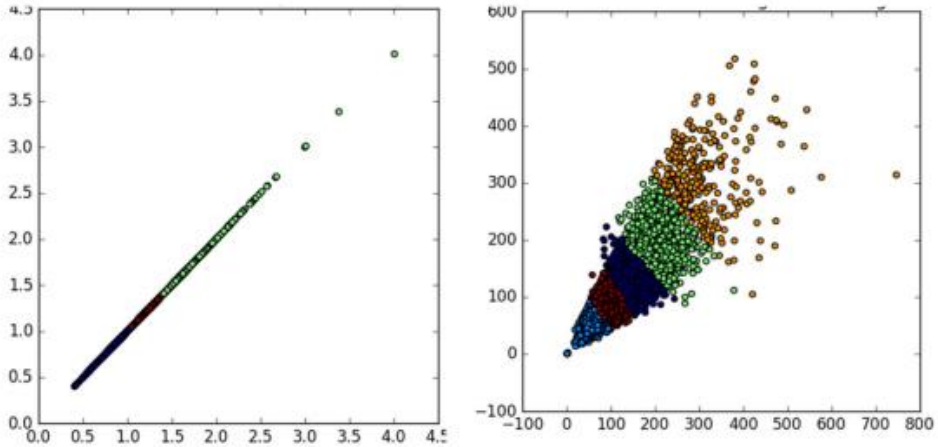
clustering on our dataset instead of choosing aspect ratios by hand. The cluster centroids are significantly different than hand-picked anchor boxes. They are suitable for vehicle detection. The k-means algorithm can be formulated as: E ¼ X k i¼1 X x∈Ci x−μi k k2 2 ð1Þ where x is the sample, μi is the average vector of Ci, and k is the center of clustering. We run k-means by various k on vehicle sizes and aspect ratios (see Fig. 2). We choose k = 5 for vehicle weight and height and k = 3 for aspect ratios as a good trade-off between accuracy and speed. It can improve 1.6% mAP on our vehicle dataset

Baseline network :

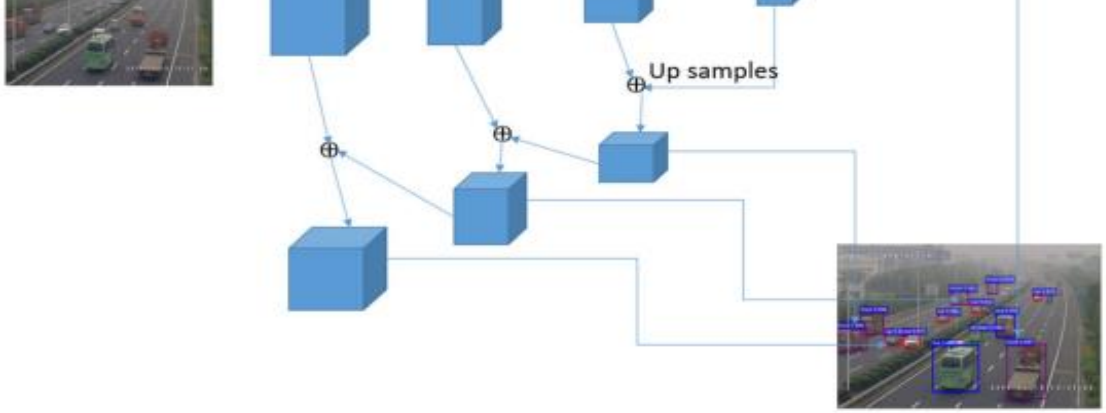
We use VGG-16 as the baseline network, which is pre-trained with ImageNet [23] dataset. It has 13 convolutional layer and three fully connected layers. In order to improve detection speed and reduce the parameters, we use convolutional layer instead of the last three fully connected layers. It has been proved to be effective in paper [8].

Feature concatenate :

Previous work on Faster R-CNN only uses the last feature map to general candidate proposal, and it is not good enough for vehicle detection, because the vehicle



scale change is larger. It is beneficial to concatenate the high-level and low-level feature [24]. The high-level feature layers have more semantic information for object classification but lack insight to precise object localization. However, the low-level feature layers have a better scope to localizing objects as they are closer to raw image. In [24], objects detect on a single concatenate feature map, and it is not accurate enough for multi-scale. In order to detect on multi-layers, we adopt feature pyramid in our network, as shown in Fig. 3. Feature pyramid can enrich the feature presentation and detect objects on different feature layers. This way is suitable for multi-scale; we can detect different size vehicles on different feature layers. As shown in Fig. 3, Firstly, a deconvolutional layer is applied to the last feature map (conv7), and a convolutional layer is grafted on backbone layer of conv6 to guarantee that the inputs have the same dimension. Then, the two corresponding feature maps are merged by element-wise addition. In our network, the last layer feature map is 5 × 5, after deconvolution is 10 × 10. The size is the same as conv6 feature map.



We use four convolutional layers (conv4–7) to generate four different size feature pyramid layers. So we can detect different size vehicles on different size feature pyramid layers. And this way can improve detection accuracy

Training and testing:

In this section, we introduce the details of network training and testing, including data augmentation, hard example mining loss function, and parameter selection.

Conclusion:

In this paper, we present an improved convolutional network for fast and accurate highway vehicle detection. We use k-means to cluster dataset and learn prior information. We use feature concatenate to extract more rich features. In order to detect different sizes of vehicles, we detect on different features. With these technology application, our framework obtains a significant improvement over Faster R-CNN and SSD, especially small vehicles. Furthermore, we will do challenging research in urban with occlusion and complex scene

Reference:

1. X. Hu et al., “SINet: A Scale-insensitive Convolutional Neural Network for Fast Vehicle Detection,” arXiv, vol 1 (2018), pp. 1–10

2. C. Stauffer and W. E. L. Grimson, Adaptive background mixture models for real-time tracking, in Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on, Fort Collins, 1999, p. 252 Vol. 2

3. G.E. Hinton, S. Osindero, Y.W. Teh, A fast learning algorithm for deep belief nets. Neural Comput. 18(7), 1527–1554 (2006)

4. S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards Real-Time Object Detection