Vehicle Classification on Low-resolution and Occluded images: A low-cost labeled dataset for augmentation

Anonymous Author(s)
Affiliation
Address
email

Abstract

Video image processing of traffic camera feeds is useful for counting and classifying vehicles, estimating queue length, traffic speed and also for tracking individual vehicles. Even after over three decades of research, challenges remain. Vehicle detection is especially challenging when vehicles are occluded which is common in heterogeneous traffic. Recently *Deep Learning* has shown remarkable promise in solving many computer vision tasks such as object recognition, detection, and tracking. We explore the promise of deep learning for vehicle detection and classification. However, training deep learning architectures require huge labeled datasets which are time-consuming and expensive to acquire. We circumvent this problem by data augmentation. In particular, we show that by properly augmenting an existing large general (non-traffic) dataset with a small low-resolution heterogeneous traffic dataset (that we collected) we can obtain state-of-the-art vehicle detection performance. This result is expected to further encourage the wide-spread use of deep learning for traffic video image processing.

1 Introduction

Traffic cameras play a crucial role in Intelligent Transport Systems. They can be used for counting vehicles, estimating queue length, traffic speed, and also for classifying and tracking individual vehicles. Here, we focus on the task of detecting and classifying vehicles from frames acquired from a traffic video stream.

Even after over three decades of research in the field, challenges remain. Vehicle detection is especially challenging when vehicles are occluded which is commonly observed in heterogeneous traffic. In heterogeneous traffic, size and type of vehicles vary significantly and vehicular traffic density is high which leads to frequent occlusion. Another issue that adds to the challenge is the low quality of the traffic camera feeds and lack of standardization of cameras and camera positions.

Traditionally, in the computer vision community, object detection is done in three steps: a sliding window phase where we search for the object at various scale and positions, followed by feature extraction at each window and finally classifying each window as either containing or not containing the desired object [3]. Commonly used features for object detection are histogram of oriented gradients (HoG) [3], scale-invariant feature transform (SIFT) [12], and speeded up robust features (SURF) [1]. This is usually followed by Support Vector Machine (SVM) based classification.

Recently, deep learning based approaches have shown extraordinary performance in many computer vision tasks such as object recognition [4], detection [16] [15], tracking [18], and image segmentation [10]. For certain tasks such as object recognition [4] and face recognition [11] deep learning has outperformed humans. The main reason behind its superior performance is, unlike traditional methods which use hand-engineered features such as HoG, SIFT, and SURF, deep networks automatically learn discriminative features from the training data directly.

In this paper, we explore the promise of deep learning for doing vehicle detection in the challenging context of heterogeneous traffic that contains significant fraction of occluded and truncated images of vehicles. Though deep learning approaches have shown state-of-the-art results for object detection, they need to be trained on huge datasets such as Imagenet [4] which has millions of images. This is because the network itself has millions of parameters to learn. However, it is very time-consuming and expensive to collect such large labeled dataset of heterogeneous traffic. The main bottleneck is the task of labeling which is required for training the deep networks. For labeling, bounding boxes need to be manually drawn around all the vehicles present in any given frame and the vehicles need to be labeled into different classes. Thus, instead of collecting a large labeled dataset for our task, we propose to use clever data augmentation techniques. We show that by augmenting a large but general (non-traffic) dataset with a small labeled traffic dataset and by training a deep network on this augmented dataset, we easily out-perform traditional approaches for vehicle detection and vehicle classification.

We collected a dataset of 1417 images from traffic cameras installed in the city of Chennai, India. This is a very small dataset to train a deep network. Thus, we have augmented the PASCAL VOC dataset [5] with our heterogeneous traffic dataset. The PASCAL VOC dataset has around 10000 images of 20 different classes including cats, dogs, trains, bottles, person along with few relevant classes such as car, truck, and bus. It is interesting to note that though PASCAL VOC has only a few relevant classes, still by augmenting it with our traffic dataset, we outperform a traditional approach of applying SIFT/SURF features followed by SVM classification. Though the proposed data augmentation can work with any deep network architecture for object detection, we have shown our results on Faster RCNN [16] which is a popular deep learning architecture.

Our specific contributions are as follows: (i) We are providing a labeled dataset for vehicle detection in heterogeneous traffic with significant occurrence of occlusion; (ii) We implement an extended deep learning architecture for the task of vehicle detection and classification in heterogeneous traffic scenario; (iii) We achieve high accuracy levels with limited data; and (iv) We demonstrate the superior performance of developed algorithm compared to a traditional object classification technique.

2 Related Work

Computer vision based methods for analyzing traffic systems are gaining in popularity. Vehicle detection and vehicle tracking have tremendously benefited from the advancements in computer vision techniques. Earlier work in vehicle detection are based on motion based algorithms (background subtraction [17], optical flow [7]) to detect vehicles and then use support vector machines [2] on the detection to classify them. [13] is one such method where authors have proposed to define a grid structure over the road in order to detect vehicles in heterogeneous traffic. These approaches are not robust with respect to illumination, occlusions, and scale changes [7] [17]. Also, the SVM classifier is heavily dependent on hand crafted features such as SURF [1] and SIFT [12].

Recently proposed deep learning models are free from these disadvantages. The most important feature of a deep learning model is: they identify useful features automatically which are quite robust to illumination and scale changes given enough training data. Authors proposed region based networks [16] [10] [9] [8] in which a network identifies possible object proposals and then a classifier classifies them. There are few studies which proposed object detection as an end to end regression problem [14] [15] [11]. All the deep learning models have been trained on huge datasets [4] which allows them to generalize well for a given task. Our method is based on one such deep learning model: Faster R-CNN (Region-based Convolutional Neural Networks) [16].

3 Methodology

Deep learning models have a large number of parameters to be tuned, which require a large number of labeled data samples. For example, in the Faster RCNN architecture, the feature extraction network (VGG-16) needs millions of high-quality images for tuning the parameters. VGG-16 is trained on Imagenet dataset [4]. The other components of Faster RCNN, region proposal network and fully

(a)

Classes	Total Samples	Occluded Samples
Light Motor Vehicles	2746	848
Heavy Motor Vehicles	279	157
Two Wheelers	3294	568

Dataset-1

Classes	Total Samples	Occluded Samples
Auto Rickshaw	598	219
Car	2148	629
Heavy Motor Vehicles	279	157
Two Wheelers	3294	568

Dataset-2

(b)

Figure 1: (a) Proposed data augmentation approach. This figure shows addition of a new class (Auto Rickshaw) from low resolution small dataset in high resolution standard dataset, (b) Statistics of our collected dataset. We have created two datasets; first one has one less class because of merging *auto-rickshaw* and *car* classes.

connected layers also require carefully annotated large datasets of images for their training. Getting such a huge labeled dataset representing every object class is very expensive and time-consuming.

Fine-tuning a pre-trained deep neural network is a standard practice in computer vision community. We have shown that doing finetuning with such a small task specific dataset performs poorly.

We test four approaches sequentially. First, the pre-trained Faster RCNN model is directly applied to our dataset. Second, the pre-trained model is fine-tuned with data from our dataset. Third, the model is trained from scratch using the collected data only. Finally, the model is trained with the existing large dataset and our collected dataset.

Our dataset is quite different from the Pascal VOC dataset on which the Faster RCNN model has been trained. In Figure 2 we have shown few of the sampled images. Pascal VOC images are high-quality images captured using high-resolution cameras whereas images in our dataset are collected from traffic surveillance camera feeds. Pascal VOC images contain fewer object instances per image compared to our dataset. One more major difference is that PASCAL VOC dataset has 20 different categories which are largely diverse. However, our dataset contains only vehicles and has different sub-categories of vehicles and vehicle classes. Due to all these differences, we can not directly deploy the existing, or even a fine-tuned, Faster RCNN model to our dataset.

Finally, we augmented the Pascal VOC dataset with our dataset. There are few vehicle classes in our dataset that are not present in Pascal VOC dataset such as auto rickshaw. We can also perform data augmentation in such a case as shown in Figure 2(b). Resultant augmented data will contain all the 20 classes of Pascal VOC and one additional class, Auto Rickshaw. While applying the model, we can ignore the irrelevant classes. This is because the model is benefiting from the high-quality images of Pascal VOC and also optimizing the loss according to our dataset. This way of augmenting the dataset with our specific dataset is leading to improved learning of parameters in the model as shown in the results section.

4 Dataset Collection

We generated our own dataset from cameras monitoring road traffic in Chennai, India. To ensure that data are temporally uncorrelated, we have sampled frames at 0.5 fps from multiple video streams. We extracted 2400 frames in total.

Figure 2: First row shows few images from Pascal VOC dataset [5]. Second row shows few images from our dataset. From these set of images it is clear that Pacsal VOC images are of higher quality compared to the images of our dataset.

Table 1: Object detection results on Faster RCNN architecture using different ways of training (AP @ 0.5).

	Model	TW	HMV	LMV
(i)	Pre-trained Model	0.256	0.273	0.600
(ii)	Pre-trained Model + Fine-tuning	0.114	0.043	0.163
(iii)	Training Only on Our Dataset	0.082	0.004	0.055
(iv)	Augmented Data Training	**0.887**	**0.968**	**0.905**

We manually labeled 2400 frames under different vehicle categories. The number of available frames reduced to 1417 after careful scrutiny and elimination of unclear images. We initially defined eight different vehicle classes commonly seen in Indian traffic. Few of these classes were similar while two classes had less number of labeled instances; these were merged into similar looking classes. For example, in our dataset, we had different categories for small car, SUV, and sedan which were merged under the light motor vehicle (LMV) category. Figure 2(b) shows brief statistics of our dataset.

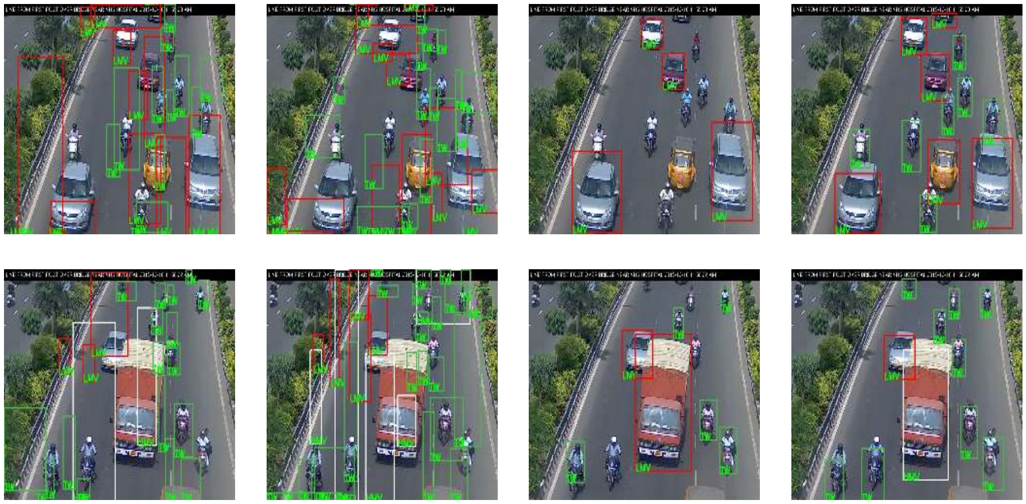
A total of 6319 labeled vehicles are available in the collected dataset (see figure 2(b)). This includes 3294 two-wheelers, 279 heavy motor vehicles (HMV), 2148 cars, and 598 auto-rickshaws. A second dataset was created by merging cars and auto-rickshaws together into light motor vehicle (LMV) class. Approximately 25.2% of vehicles were occluded.

We have released the heterogeneous traffic dataset that we collected[1] for public use.

5 Experimental Results

In this section, we show the results of proposed data augmentation approach and performance obtained by extending faster RCNN model for new classes. The results of data augmentation are compared with the performance of four different ways of training Faster RCNN on our dataset: (i) training from scratch using collected dataset alone, (ii) fine-tuning the pre-trained model with collected dataset, and (iii) using pre-trained model directly, and (iv) model trained from scratch using augmented dataset. Performance of extended Faster RCNN model is compared with three different ways of training: (i) using pre-trained Faster RCNN model for object proposals alone and then using SVM for

[1] https://www.dropbox.com/s/j1gr0d4w8u57jfv/dataset_vehicle_detection_ilds_iitm.tar.gz?dl=0



(a) (b) (c) (d)

Figure 3: Vehicle detection results on Dataset-1. (a) Faster RCNN model fine-tuned on our data, (b) Faster RCNN model trained on our data from scratch, (c) Faster RCNN pre-trained on PASCAL VOC data, (d) Extended Faster RCNN model trained on 3-class augmented data.

Table 2: Results on adding a new class to the model (AP @ 0.5).

	Model	AR	TW	HMV	LMV
(i)	Pre-Trained Model + SVM	0.195	0.132	0.417	0.58
(ii)	Data augmentation (Dataset-1) + SVM	0.609	0.783	0.653	0.87
(iii)	Data Augmention (Dataset-2)	**0.983**	**0.883**	**0.987**	**0.905**

classification, (ii) training Faster RCNN on augmented dataset and then using SVM for classification and (iii) extending Faster RCNN with a new class: auto-rickshaw.

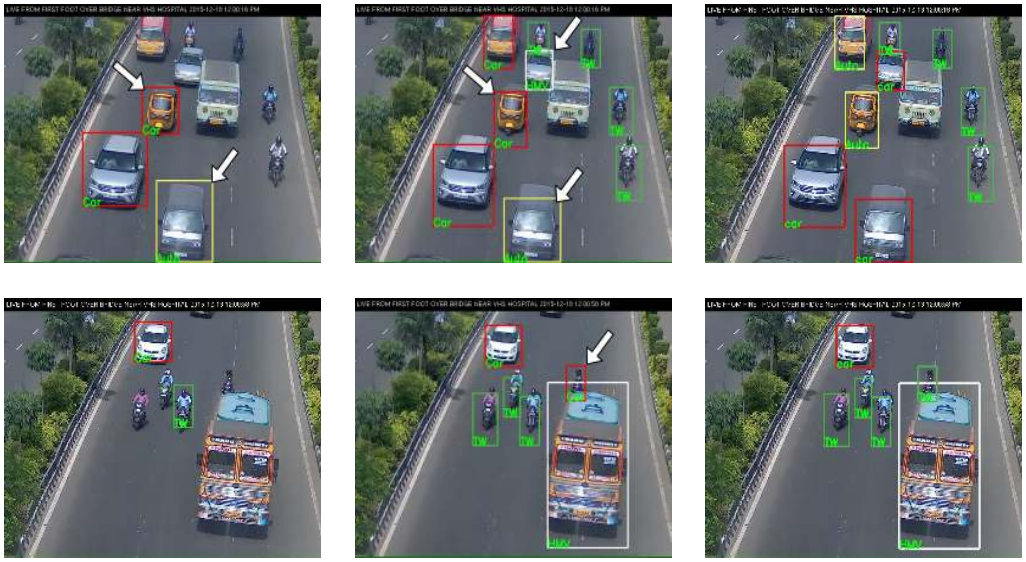
All the experiments have been performed on a machine with dual core Intel Xeon processor (2.20 GHz) having 256 GB of DDR4 RAM with one TitanX graphics processing unit (GPU). Using Faster RCNN model we achieved processing speed at 5 frames per second.

5.1 Data augmentation

Table 1 shows results of Faster RCNN architecture using different types of training on Dataset-1 that has three classes: 1) Two wheelers (TW), 2) Light Motor Vehicle (LMV), and 3) Heavy Motor Vehicle (HMV). From this table, we can infer that pre-trained model gives poor results. The poor performance of the fine-tuned model can be attributed to the difference in quality and content of the collected data compared to Pascal VOC. The model trained only on the collected dataset is performing poorly because of limited data. The model trained from scratch on augmented data is performing best because it is learning from both datasets; it is benefiting from the good features present in Pascal VOC dataset and also optimizing parameter values according to our dataset. Image outputs from each approach are shown in Figure 3.

5.2 Extending Faster RCNN Model for new classes

To compare deep learning approaches with traditional approaches we have trained different SVM models for vehicle classification. In order to generate feature vectors for SVMs' training, we extracted SIFT features from the image patches, where each patch contains only one vehicle. Once we have cropped a patch from an image, we change its color space from RGB to gray-scale. Then, SIFT and SURF features are extracted for all the patches. K-means clustering is done separately on the SIFT and SURF features extracted from all the patches. The final feature vector for each patch is then



(i) Pre-trained Model+SVM (ii) Data Augmentation+SVM (iii) Extended Faster RCNN

Figure 4: Image outputs of extended Faster RCNN model on Dataset-2.

generated following bag-of-words [6] approach*, i.e.,* for a given 'k' we compute a number of the SIFT features getting assigned to a particular cluster center. Experiments were done with different values of 'k'. Similarly, another feature vector corresponding to SURF features was generated. After getting these feature vectors, SVMs were trained on top of it.

As explained in the dataset section, we have merged the eight vehicle categories into three vehicle categories. This allowed us to make the best use of Faster RCNN architecture with the existing pre-trained model with minimum modifications. The results are shown in Table 2. Faster RCNN architecture is able to detect all vehicles well; however, it is unable to classify auto-rickshaw since it is not trained on our data. One solution is to train an SVM model to do the classification instead. Therefore, in this setting, we get the object proposals from the Faster RCNN model to detect vehicles and then employ SVM to classify the detected vehicles into different classes. Finally, we extended the Faster RCNN model to incorporate a new class. Adding a new class in Faster RCNN model and then training with augmented data gives the best results. Image outputs from each model are shown in Figure 4.

6 Conclusion

Deep learning has emerged as a significant new paradigm in object identification and classification. However, training deep learning networks requires large datasets. In this paper, we demonstrate the use of a limited traffic dataset that augments existing large scale datasets and uses an existing deep learning network (Faster RCNN) for detecting and classifying vehicles several of which are truncated or occluded. The extended faster RCNN model is also able to deduct a new class of vehicles with high degree of accuracy. The results obtained are promising for heterogeneous traffic scenario where occlusion is common. This result is expected to encourage the wide-spread use of deep learning for traffic video image processing since it is economical in terms of cost and time.

The results open up significant avenues for further research. For example, the present model works at 5 fps on TitanX GPU because of the high computation time of Faster RCNN. To make this model run in real-time is one future work direction. A larger dataset with more instances of each class can be used to train an eight- or ten-vehicle class model. Given the dissimilarities particularly among vehicle types grouped under heavy vehicles, such a finer classification may result in significant improvements to overall accuracy. Testing the robustness of developed models with multiple video inputs with varying environmental parameters is on-going.

References

[1] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. *Computer vision–ECCV 2006*, pages 404–417, 2006.

[2] Olivier Chapelle, Patrick Haffner, and Vladimir N Vapnik. Support vector machines for histogram-based image classification. *IEEE transactions on Neural Networks*, 10(5):1055–1064, 1999.

[3] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009.

[5] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[6] AG Faheema and Subrata Rakshit. Feature selection using bag-of-visual-words representation. In *Advance Computing Conference (IACC), 2010 IEEE 2nd International*, pages 151–156. IEEE, 2010.

[7] David Fleet and Yair Weiss. Optical flow estimation. In *Handbook of mathematical models in computer vision*, pages 237–257. Springer, 2006.

[8] Ross Girshick. Fast r-cnn. In *International Conference on Computer Vision (ICCV)*, 2015.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *European Conference on Computer Vision*, pages 346–361. Springer, 2014.

[10] Yi Li, Kaiming He, Jian Sun, et al. R-fcn: Object detection via region-based fully convolutional networks. In *Advances in Neural Information Processing Systems*, pages 379–387, 2016.

[11] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*, pages 21–37. Springer, 2016.

[12] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[13] Gitakrishnan Ramadurai Manipriya, S. and VV Bhavesh Reddy. Grid-based real-time image processing (grip) algorithm for heterogeneous traffic. In *In Communication Systems and Networks (COMSNETS), 2015 7th International Conference on Intelligent Transportation Systems*, pages 1–6. IEEE, 2015.

[14] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016.

[15] Joseph Redmon and Ali Farhadi. Yolo9000: Better, faster, stronger. *arXiv preprint arXiv:1612.08242*, 2016.

[16] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2015.

[17] S Cheung Sen-Ching and Chandrika Kamath. Robust techniques for background subtraction in urban traffic video. In *Electronic Imaging 2004*, pages 881–892. International Society for Optics and Photonics, 2004.

[18] Lijun Wang, Wanli Ouyang, Xiaogang Wang, and Huchuan Lu. Stct: Sequentially training convolutional networks for visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1373–1381, 2016.