Vehicle Detection and Re-Identification

Ibrahim Soliman[1,a] , KC Lim[1,b]

Faculty of Electronic & Computer Engineering, Universiti Teknikal Malaysia Melaka
ibrahimsoliman97@gmail.com[a] , kimchuan@utem.edu.my[b]

ABSTRACT

This paper proposes an approach to the vehicle detection, tracking and re-identification problem in a surrounding area from surveillance car. In our project, we implement, and test several state-of-the-art detection trained on domain general datasets. We experiment with different levels of transfer learning for fitting these models over to our domain. We report our result of cascading YOLOv2[1] as a vehicle detection with Deep SORT[2] as a pragmatic approach to multiple vehicle tracking with a focus on simple, effective algorithms with integration of appearance information. Our pre-result shows an impressive accuracy in tracking and re-identification of multiple vehicles through long periods of occlusions in the input stream video with an acceptable frame rate of 10 FPS.

INTRODUCTION

Vehicle search and re-identification is an important problem in computer vision, which has many practical applications like driver assistance, intelligent parking, self-guided vehicle systems, and traffic monitoring (quantity, speed, and flow of vehicles. Although the license plate provides a unique ID for a vehicle, sometimes it is still not easy to recognize its plate. For example, the resolution of images is not enough due to the environment or the camera, or the plate is occluded or removed. Thus, vehicle re-identification based on appearance information still plays an important role for real applications. Although vehicle identification problem is of a great importance, most previous object identification works focus on human face or person [4, 5, 6, 7]. However, their targets are similar, which are to learn discriminative representations for images.

Recently, deep convolutional network has also demonstrated its great power in identification tasks. In [4], Yi et al. introduced a deep network to directly classify all identities (about 10,000 classes) for face recognition. Then, a pair-wise verification loss [5] is proposed to be combined with identification loss to help reduce intra-class variations by pulling features of same identity together. Similar verification loss is also utilized in person identification. Another successful deep learning framework is triplet loss for both face recognition and person re-identification [7]. It learns the embedding representations in the deep convolution network by optimizing the triplet loss, which is under the assumption that samples of the same identity should be closer from each other than samples of different identities.

Different with face or person identification problem, vehicle identification could be more challenging since it is really hard to discriminate vehicles with similar visual appearance which belongs to the same model. Most previous related works about vehicle focus on the vehicle model classification [8, 9] which only recognize vehicle models instead of further identities. Recently, Liu et al. [3] presented a new large-scale vehicle re-identification database ‘VehicleID’, which is collected from the real surveillance cameras and labeled in identity level. The large scale of the dataset facilitates the recent deep learning models, which have been proved more effective and robust for many vision tasks, to apply to the vehicle identification problem. Inspired by some state-of-the-art methods in face recognition [6], in this paper, we propose a cascading of vehicle detection with tracking and re-ID model.

The proposed model is an end-to-end multi-task deep framework, which aims to learn a deep convolutional model that can extract discriminative features for vehicle images. The overall network is illustrated in Figure 1. Our cascaded networks incorporate two different networks in a unified framework. The first network includes multi vehicle detection using YOLOv2. While second network is Deep SORT that preforms a tracking task using Kalman filter with Hungarian method and using CNN as a features extractor for re-identification task.

METHODOLOGY

The proposed method is illustrated in Figure 1, and is based on the two main steps. First, the streamed video is passed to YOLOv2 as an input to provide a detection and localization of the vehicle in this video. These regions are taken as the likely locations to inspect for cars since one region—or a small group of them—may represent a car. The second step is devoted to the feature extraction process and tracking filter, where a window around the candidate region is given as input to a pre-trained CNN for feature extraction and imaging processing filter, this algorithm that combine the CNN feature extractor and Kalman filter with Hungarian method called Deep SORT .

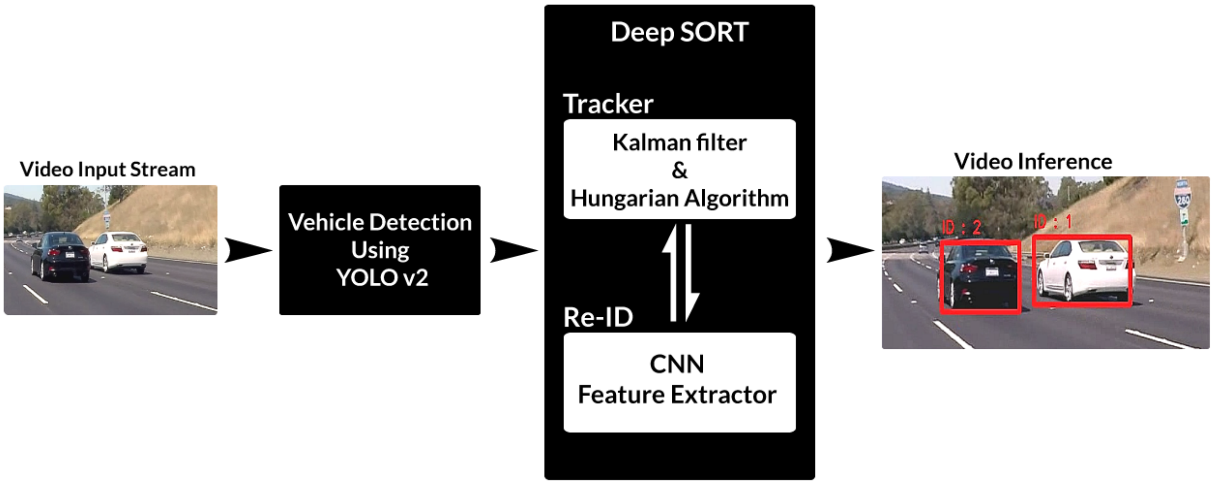


Figure 1, Architecture and flow of the proposed cascaded networks

1. *Vehicle Detection*

"You Only Look Once (Yolo)" is a desktop application for the real-time object detection system. Object detection system from the application repurposes classifiers or localizers to perform detection. The application work by applying the model to an image at multiple locations and scales. The system will consider high scoring regions as detections. A single neural network divides the image into regions and predicts bounding boxes and class probabilities directly from full images in one evaluation. In figure 2.1, a full comparison between the most popular object detection that has recently designed.

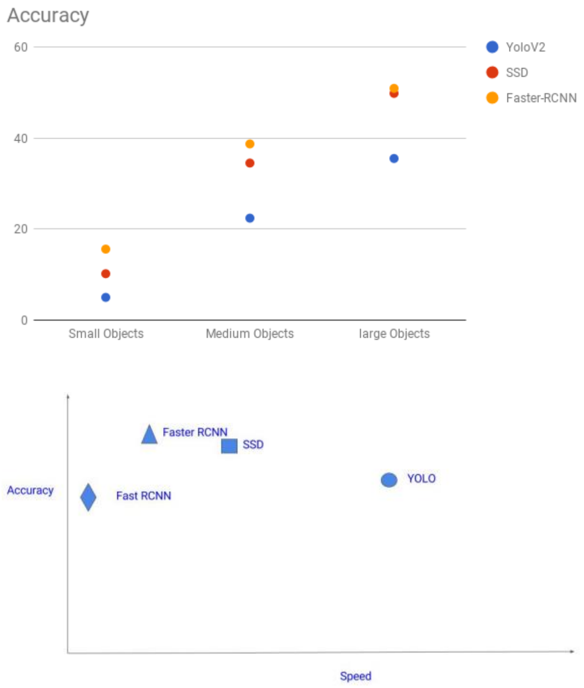


Figure 2.1, Comparison between YOLO, SSD, Faster RCNN[13]

2. *Deep SORT*

Simple online and real-time tracking (SORT) [10] is a much simpler framework that performs Kalman filtering in image space and frame-by-frame data association using the Hungarian method with an association metric that measures bounding box overlap. This simple approach achieves favorable performance at high frame rates. On the MOT challenge dataset [11], SORT with a state-of-the-art people detector ranks on average higher than MHT on standard detections. This not only underlines the influence of object detector performance on overall tracking results, but is also an important insight from a practitioner's point of view.

While achieving overall good performance in terms of tracking precision and accuracy, SORT returns a relatively high number of identity switches. This is, because the employed association metric is only accurate when state estimation uncertainty is low. Therefore, SORT has a deficiency in tracking through occlusions as they typically appear in frontal-view camera scenes. To overcome this issue a replacing of association metric with a more informed metric that combines motion and appearance information. In particular has been performed, by applying a convolutional neural network (CNN) that has been trained to discriminate pedestrians on a large-scale person re-identification dataset.

By using simple nearest neighbor queries without additional metric learning, successful application of this method requires a well-discriminating feature embedding to be trained offline, before the actual online tracking application. To this end, we employ a CNN that has been trained on a large-scale person re-identification dataset [12] that contains over 1,100,000 images of 1,261 pedestrians, making it well suited for deep metric learning in a people tracking context. The CNN architecture of DeepSORT network is shown in appendix A. In summary, a wide residual network with two convolutional layers followed by six residual blocks has been applied. The global feature map of dimensionality 128 is computed in dense layer 10. final batch and l_2 normalization projects features onto the unit hypersphere to be compatible with our cosine appearance metric.

Two effective parameters that play an important role in DeepSORT performance:

i) N_init: is a number of consecutive detections before the track object is confirmed , the track state is set to 'Deleted' if a miss occurs within the 'n_init' frames.
ii) Max_age: the maximum number of consecutive misses of vehicle in the frame before the track state of the feature in the tracking list is set to 'Deleted'.

The selected value for N_init is 2 while the Max_age is set to 8 frames. In total, the DeepSORT network has 2,800,864 parameters. One forward pass of the DeepSORT, with 32 bounding boxes produced by YOLOv2, takes approximately 32 ms on an Nvidia Quadro M4000 GPU. Thus, the selected network is well suited for online tracking, provided that a modern GPU is available.

RESULTS & EXPERIMENT

We have tested our network on a various set of videos on Nvidia Quadro M4000 GPU. Our cascaded network reaches 9 FPS for inference performance on stated GPU as shown in table 2. And we can increase the FPS by skipping some frames from video stream that will not effect on accuracy. For skip one frame, the inference reaches 18 FPS.

	YoloV2 (standalone)	DeepSORT (standalone)	Yolo+DeepSORT (Overall)
FPS	13	23~25	8~9

Table 1, Tracking Network frame rate per second results.

By using VOTT "Visual Object Tagging Tool" from Microsoft, a groundtruth video annotation test-set has been created to measure the MOTA(1) (Multiple Object Tracking Accuracy) and MOTP(2) (Multiple Object Tracking Precision)[14] for different video test-sets .

$$MOTA = 1 - \frac{\Sigma_t\ (m_t + fp_t + \mathrm{mme}_t\ )}{\Sigma_t\ g_t} \tag{1}$$

Where m_t , fp_t and mme_t are the number of misses, of false positives and of mismatches respectively for time t.

$$MOTP = \frac{\Sigma_{i,t}\ \mathrm{d}_{i,t}}{\Sigma_t\ c_t} \tag{2}$$

Where c_t denotes the number of matches in frame t and $d_{t,i}$ is the bounding box overlap of target I with its assigned ground truth object.

	MOTA	MOTP
180° Camera (non 360° cam, 50sec @ 25FPS)	95.3 %	90.5 %
GIROPTIC 360°Cam (low resolution 1280x1040, 300 sec @ 25FPS)	77.6 %	73.3 %
360° High resolution Camera (1920x1080, 180 sec @ 30FPS)	91.1 %	84.5 %

Table 2, Proposed Network results using different types of cameras.

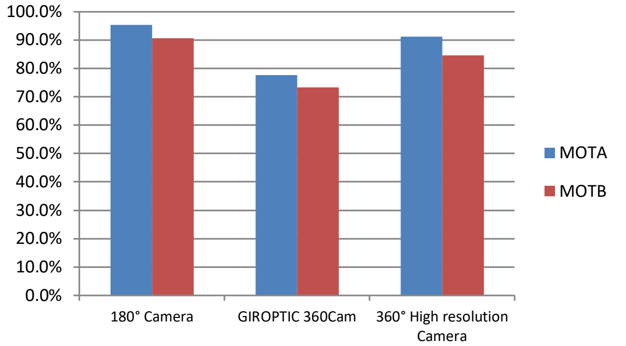


Figure 3, Proposed Network results using different types of cameras.

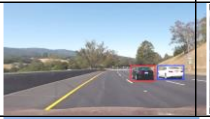
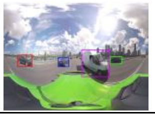
	180° Camera	GIROPTIC 360°Cam	360° High resolution Camera
Ground truth			
Inference			

Table 3, Compression between groundtruth and inference for different frames from different types of videos.

A inference video have uploaded on YouTube in the following link: https://youtu.be/oftvj5--1cI

CONCLUSION

We presented a video based vehicle re-identification system which based on the popular object detection network 'YOLOV2' as vehicle detection network and cascaded with the extension of SORT, the DeepSORT, for vehicle re-identification. By adapting the parameters, we are able to track vehicle through longer periods of occlusion that can be used in various application. In future work, retraining YOLOv2 in low resolution video dataset can accomplish higher MOTA and MOTP.

REFERENCES

[1] J.Redmon and A.Farhadi, YOLO9000: Better, Faster, Stronger (2016) arXiv:1506.02640v5.

[2] A. Bewley, G. Zongyuan, F. Ramos, and B. Upcroft, “Simple online and realtime tracking,”.ICIP, 2016.

[3] Hongye Liu, Yonghong Tian, Yaowei Yang, Lu Pang, and Tiejun Huang, “Deep relative distance learning: Tell the difference between similar vehicles,” in CVPR, 2016.

[4] Yi Sun, Xiaogang Wang ,and Xiaoou Tang, “Deeplearning face representation from predicting 10,000 classes,” in CVPR, 2014.

[5] Yi Sun, Yuheng Chen, Xiaogang Wang, and Xiaoou Tang, “Deep learning face representation by joint identification-verification,” in NIPS, 2014.

[6] Yi Sun, Xiaogang Wang, and Xiaoou Tang, “Deeply learned face representations are sparse, selective, and robust,” in CVPR, 2015.

[7] Florian Schroff, Dmitry Kalenichenko, and James Philbin, “Facenet: A unified embedding for face recognition and clustering,” in CVPR, 2015.

[8] Yen-Liang Lin, Vlad I Morariu, Winston H Hsu, and Larry S Davis, “Jointly optimizing 3d model fitting and fine-grained classification.,” in ECCV, 2014.

[9] Edward Hsiao, Sudipta NSinha, Krishnan Ramnath, Simon Baker, Larry Zitnick, and Richard Szeliski, “Car makeandmodelrecognitionusing3dcurvealignment,” in WACV, 2014.

[10] A. Bewley, G. Zongyuan, F. Ramos, and B. Upcroft, “Simple online and realtime tracking,” in ICIP, 2016, pp. 3464–3468.

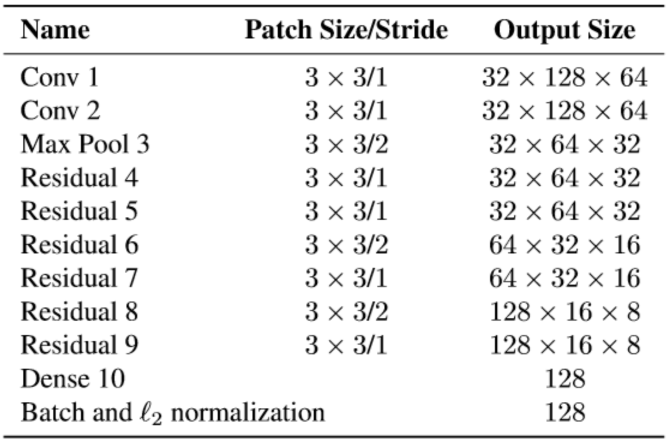
[11] L. Leal-Taix´e, A. Milan, I. Reid, S. Roth, and K. Schindler, “MOTChallenge 2015: Towards a benchmark for multi-target tracking,” arXiv:1504.01942 [cs], 2015.

[12] L. Zheng, Z. Bie, Y. Sun, J. Wang, C. Su, S. Wang, and Q. Tian, “MARS: A video benchmark for large-scale person re-identification,” in ECCV, 2016.

[13] Zero to Hero: Guide to Object Detection using Deep Learning: Faster R-CNN,YOLO,SSD Retrieved from http://cv-tricks.com/object-detection/faster-r-cnn-yolo-ssd/

[14] Keni Bernardin, Alexander Elbs, Rainer Stiefelhagen: Multiple Object Tracking Performance Metrics and Evaluation in a Smart Room Environment,2016.

Appendix 1



Name	Patch Size/Stride	Output Size
Conv 1	$3 \times 3/1$	$32 \times 128 \times 64$
Conv 2	$3 \times 3/1$	$32 \times 128 \times 64$
Max Pool 3	$3 \times 3/2$	$32 \times 64 \times 32$
Residual 4	$3 \times 3/1$	$32 \times 64 \times 32$
Residual 5	$3 \times 3/1$	$32 \times 64 \times 32$
Residual 6	$3 \times 3/2$	$64 \times 32 \times 16$
Residual 7	$3 \times 3/1$	$64 \times 32 \times 16$
Residual 8	$3 \times 3/2$	$128 \times 16 \times 8$
Residual 9	$3 \times 3/1$	$128 \times 16 \times 8$
Dense 10		128
Batch and ℓ_2 normalization		128

The CNN architecture of DeepSORT network