



# Faster CNN-based vehicle detection and counting strategy for fixed camera scenes

Ahmed Gomaa<sup>1</sup> · Tsubasa Minematsu<sup>2</sup> · Moataz M. Abdelwahab<sup>3</sup> · Mohammed Abo-Zahhad<sup>3,4</sup> · Rin-ichiro Taniguchi<sup>2</sup>

Received: 14 April 2021 / Revised: 19 June 2021 / Accepted: 18 January 2022 /  
Published online: 23 March 2022  
© The Author(s) 2022

## Abstract

Automatic detection and counting of vehicles in a video is a challenging task and has become a key application area of traffic monitoring and management. In this paper, an efficient real-time approach for the detection and counting of moving vehicles is presented based on YOLOv2 and features point motion analysis. The work is based on synchronous vehicle features detection and tracking to achieve accurate counting results. The proposed strategy works in two phases; the first one is vehicle detection and the second is the counting of moving vehicles. Different convolutional neural networks including pixel by pixel classification networks and regression networks are investigated to improve the detection and counting decisions. For initial object detection, we have utilized state-of-the-art faster deep learning object detection algorithm YOLOv2 before refining them using K-means clustering and KLT tracker. Then an efficient approach is introduced using temporal information of the detection and tracking feature points between the framesets to assign each vehicle label with their corresponding trajectories and truly counted it. Experimental results on twelve challenging videos have shown that the proposed scheme generally outperforms state-of-the-art strategies. Moreover, the proposed approach using YOLOv2 increases the average time performance for the twelve tested sequences by 93.4% and 98.9% from 1.24 frames per second achieved using Faster Region-based Convolutional Neural Network (FR-CNN) and 0.19 frames per second achieved using the background subtraction based CNN approach (BS-CNN), respectively to 18.7 frames per second.

**Keywords** Vehicle detection and counting · Convolution neural network · Fixed camera · Faster R-CNN · YOLOv2 · Feature point analysis

## 1 Introduction

Traffic information analysis is an important task in an intelligent transportation system (ITS) [5, 38] by offering efficient information for traffic control and management. Estimat-

---

✉ Ahmed Gomaa  
ahmed.gomaa@nriag.sci.eg

ing the number of vehicles running in video sequences is an important process in various applications used in selecting the best routes, manage the traffic lights, and help governments to make a decision about building new roads and plan the expansion of the traffic system. These numbers indicate the traffic status, including congestion level, lane occupancy, and road-traffic intensity [8]. Such kind of information can be utilized for automatic route planning, prevent road congestion, and early incident detection. In the traditional intelligent transportation systems, vehicle counting process is always implemented using special sensors. However, their higher installation cost and the simple format represent some limitations for those sensors.

With the growth of digital video processing, counting system based on image processing and computer vision techniques, offers an attractive alternative method for its large ability in detecting vehicle type, velocity, density, and road traffic accident [3]. The machine vision vehicle counting method is an integrated procedure comprised of detection, tracking, and trajectory processing. Vehicle detection is considered the first step in obtaining traffic flow characteristics [34]. Its purpose to obtain the location and classification of the object from an image [13]. Its primary task is to acquire the features of the object. Traditional detection approaches based on low ranks decomposition [31, 37], gaussian mixture modeling [40], Morphological operations [12], and principal component analysis [24] can be used for the detection. It exhibits better performance if the object has some deformation or scale change. However, these methods cannot adapt to large rotations, has poor stability, and is slow to calculate. In recent years, Deep Learning strategies, more specifically convolutional Neural Networks (CNNs) [10, 11, 17, 20, 25], have shown significant advances over the traditional approaches in many computer vision tasks, including object detection and classification.

Detection and tracking of vehicles are normally done as two separate procedures. Vehicle detection in images relies on features of spatial appearance, while vehicle tracking relies on features of spatial appearance as well as temporal motion [6, 9, 13]. Vehicle detection techniques focused on visual and appearance characteristics of vehicles' spatial domain have begun to saturate. Different methods have been introduced in the literature to address the detection and counting problems of vehicles in images and videos. However, due to the changes in shape, scale, object view, shadows, lighting conditions, and partial occlusion, it is still an open issue. Although substantial advances have been made separately for detection or tracking, most of them still require a high computation complexity with low accuracy.

Not much effort has been made to use tracking information to increase object detection and counting accuracy and minimize storage and training costs. In addition, the object detector usually suffers from missed detection and false positives that deteriorate the process of counting. Due to such issues, this work proposes a successful collaborative strategy between detection and tracking information to strengthen both detection and counting processes. Detection and tracking of vehicles simultaneously are considered an efficient technique for achieving precise detection and counting results. Firstly, we use the transfer learning for the YOLO-v2 vehicle detector as an initial step to discriminate between vehicles and other foreground and background regions. The second point of our contribution is to refine the detection result to achieve a perfect vehicle detection result, the optical flow tracking information of the detected feature points is used to achieve an accurate vehicle detection. The third part of our contribution is to assign each vehicle with its corresponding trajectory based on the temporal information of the refinement detection and tracking feature points between the framesets to achieve better detection and counting results.

Moreover, to enhance both the time performance and the overall detection and counting accuracy, different CNN-based detections, including BS-based CNN, FR-CNN, and YOLO

v2, were tested in the detection part with the tracking information help. These methods' overall performance is analyzed in terms of detection, counting, and time performance accuracy. Also, to highlight our contribution about the importance of using both the detection and tracking information in the detection decision, we evaluated the detection accuracy in two cases

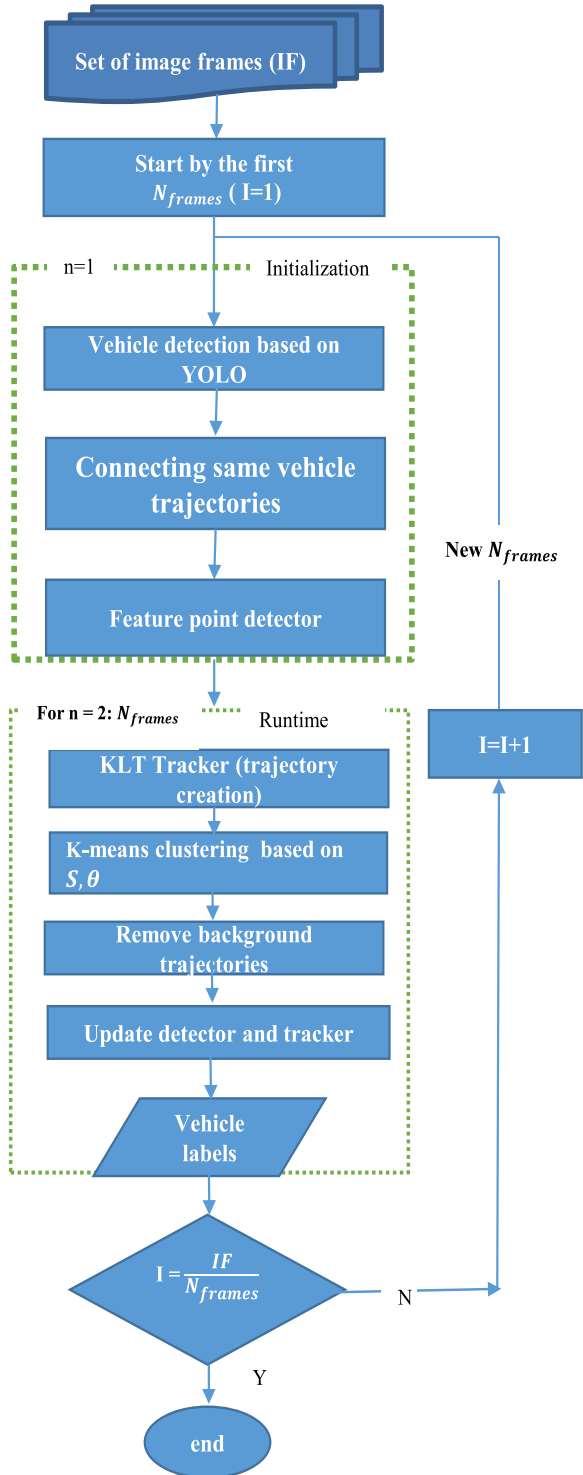
- Detection decision is based only on the YOLO-v2 and the detection implemented on every frame.
- Detection decision is based on the YOLO-v2 and tracking information, and the detection is implemented once on the first frame, every fixed number of frames.

This work used the detection information of YOLOv2 based CNN, and tracking information using optical flow. The first step contains vehicle detection and features extractor, while the second step to analyze these features and vehicle refinements. First, the power of the convolution neural network is exploited in the vehicle detection process before the vehicle refining and clustering process in the second step using the optical flow and k-means clustering. The CNN is used in the first frame, with the refining analysis considering the remaining frames in the frameset. Thus, a robust discrimination process between the foreground vehicles and noisy background regions is utilized. Thirdly, an effective counting strategy is offered to assign each vehicle with its corresponding trajectory based on the collected detection and tracking information. Moreover, we consider the performance evaluation of three categories of CNN architectures in the context of vehicle detection, in terms of accuracy and processing time. We conducted experiments on challenging datasets, and the proposed method showed the best performance in terms of precision and recall. The proposed framework contains three functional steps in each frameset, as shown in Fig. 1. In summary, this study provides the following four main contributions:

- Rapid and reliable vehicle detection and counting strategy is presented. The developed strategy exploits the regression based CNN benefits and the optical flow information to obtain a faster and more reliable result.
- Comprehensive analysis of the performance of three categories of CNN architectures used in the detection part in cooperate with tracking information in the context of vehicle detection and counting is presented, in terms of accuracy and processing speed.
- The performance evaluation of two detection based categories is presented, the first one when the detection decision is based only on the YOLO-v2 and the second when the detection decision is based on both the YOLO-v2 and tracking information.
- Extensive experiments on four benchmark datasets demonstrate the effectiveness of the proposed strategy and its ability to achieve faster detection and counting results with efficient accuracy.

The main steps of proposed vehicle detection and counting scheme are described in the following subsections. The rest of this work is organized as follows. Section 2 summarizes the related work. Section 3 presents the proposed strategy. Section 4 reports the experimental results and discussion. Section 5 demonstrates the conclusion.

**Fig. 1** Architecture of the proposed system (n is the frame number in video)



## 2 Related work

In recent years, several deep learning networks have been utilized in vehicle detection and counting. A classification framework based on deep convolutional neural network (DCNN) was previously reported. Its accuracy has been greatly improved compared with traditional classification algorithms, which lays a foundation for deep learning-based object detection research. However, only detection information has some false positive and negative results. Ross Girshick et al. used a selective search algorithm named regions with CNN (RCNN) based on region proposal strategy that searches for possible object regions [11]. According to the slow detection speed of RCNN, Ross Girshick et al. proposed Fast RCNN [10], by adding bounding box regression and multi-task loss function. In [27], the authors added a new region proposal network based on the fast RCNN algorithm and proposed the Faster RCNN. The accuracy of the Faster RCNN has been greatly improved and is rated the best in all current detection algorithms, but the speed is one of its drawbacks. To solve this problem, Liu proposed an end-to-end detection algorithm, a single-shot multi-box detector (SSD) [20], which obtains proposal regions by uniform extraction and greatly enhances the detection speed.

In the most recent year, Redmon et al. proposed YOLOv2 [25], and YOLOv3 [26] by using multi-scale prediction and improving the basic classification network, with fast detection speed. In [1], authors present YOLOv4 that utilizes an additional bounding box regressor based on the Intersection over Union (IoU) and a cross-stage partial connections in their backbone architecture, and used mosaic and cut-mix data augmentation. However, all of the mentioned algorithms depends only on the detection decision each frame, so the accuracy is still needs improvement with fast processing speed approach.

Authors in [4], presented a new vehicle detection and counting strategy by employing a real-time background model, filling holes and denoising optimization, and motion-based information analysis. This strategy can handle vehicle shadows and sudden illumination changes. Moreover, the counting accuracy has been enhanced by using the ROI concept called Normative-Lane and Non-Normative-Lane. However, their use of traditional background modeling and the morphological operation will affect the results of the algorithm and deteriorate the detection and counting performance. Song et al. proposed a vehicle detection and counting system by employing YOLOv3 for detecting the vehicles. Then, the ORB approach [28] was adopted for the vehicle trajectories and counting [33]. This algorithm achieved a satisfying accuracy, but it still needs more time improvement because it depends on the feature point purification to exclude the false noise points using the RANSAC algorithm by estimated the homography matrix. Authors in [30], present a combined detection and tracking system that uses YOLOv3 for object detection and the Deep SORT for object tracking. They achieved better detection and counting accuracy, but their algorithm still requires more time.

Recently, the Convolutional neural network has achieved high performance for object detection, while dealing with CNN efficiently for real-time vehicle counting is still a challenging problem. In [14], authors combined the detection based CNN decision and tracking information to solve this problem. The algorithm firstly detects the vehicles using a CNN-based classifier with connected component labeling, then vehicle feature motion is analyzed to remove the noise and cluster the vehicles. Vehicle refinement has been done using a new combined strategy between the K-means clustering and optical flow tracking information. Finally, a way to assign the detected vehicles with its corresponding cluster is introduced, to ensure a non-repeated counting process, by considering the intersection area between

the detected and tracked point information. However, this algorithm achieved a promising result using the previously mentioned technique, but the processing time still needs to be reduced since the strategy employing a pixel classification CNN-based strategy for the detection. Various background subtraction based CNN strategies are introduced to overcome the quite slow for the patch-wise based methods by considering the full image as an input [21, 39]. However, background subtraction strategies cannot classify and differentiate the object class including vehicles, bicycles, or persons. So in this work, we evaluate different object detection and classification strategies like Faster R CNN and YOLOv2, with the tracking information help because we focus on vehicles in this work.

As concluded from the evaluation process, we present an efficient and fastest strategy for vehicle detection and counting. The proposed algorithm detects moving vehicles based on YOLOv2 classifier. Then, the vehicle's robust features are refined and clustered by motion feature points analysis using a combined technique between KLT tracker and K-means clustering. Finally, an efficient strategy is presented using the detected and tracked points information to assign each vehicle label with its corresponding one in the vehicle's trajectories and truly counted it.

### 3 Proposed methodology

The proposed method divided into three steps. First, Vehicles are classified and detected every N-frames using YOLOv2. Then, their features are extracted and analyzed by tracking and clustering the corner points through the N-frames using K-means clustering and KLT tracker. Finally, an efficient strategy is presented to assign vehicle trajectories to each detected bounding box, so that each vehicle trajectories will take a unique label. The architecture of the proposed approach is shown in Fig. 1. The main steps of proposed vehicle detection and counting scheme are described in the following subsections.

#### 3.1 Vehicle detection

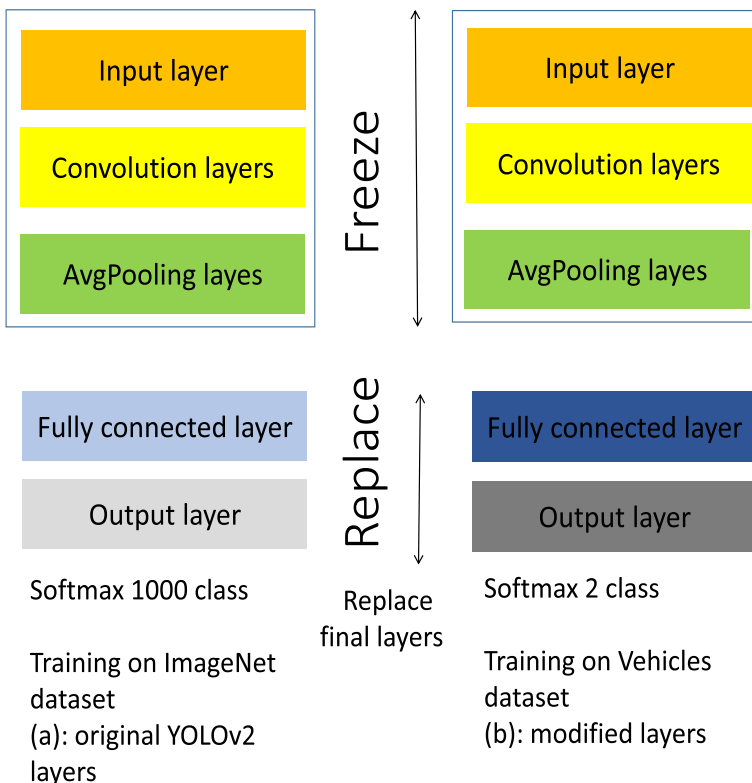
Recently, different deep learning-based strategies are used to enhance the detection result such as Single Shot MultiBox Detector (SSD) [20], region proposal approaches [11, 27], and You Only Look Once (YOLOv2) [25]. However, authors in [14] achieved a promising result using background subtraction based CNN for the detection with the help of tracking information, but the processing time still needs to be reduced since they used a pixel classification CNN-based strategy for the detection.

Recently, It is well known that YOLOv2 is one of the fastest object detection methods based on CNN compared to the state of the arts [23], so in this work, we investigate YOLOv2 in the detection phase combined with the tracking information to achieve faster detection and counting strategy. YOLOv2 is trained on more than a million images from the ImageNet database that consists of 1.2 million images classified into 1000 classes [29]. In our work, the YOLOv2 layers are used up to the last fully connected layer that changed from 1000 classes to two classes as we focus only on vehicles detection.

In this work, we exploit the Yolov2 architecture to achieve faster detection result employing the transfer learning. In addition using the optical flow information to improve the detection performance with the same counting strategy explained in [14] to achieve perfect counting decision. Transfer learning is implemented on the final layers by replacing the softmax 1000 classes by softmax two classes. In transfer learning, pre-trained convolution

neural network models are used which trained using big datasets. pre-trained model layers are used up to the last fully connected layer that is trained in our work using the dataset associated with vehicles. More details of the transfer learning method appear in the work of [22]. Since we utilize the transfer learning in this work, we used the Resnet-50 [16] as a pertained neural network model and considered a backbone for the YOLOv2. A block diagram illustrating the ResNet50 transfer learning architecture is shown in Fig. 2. ResNet50 is a CNN that trained on more than a million images from the ImageNet database. This database contains 1.2 million images classified into 1 thousand classes. Three training experiments were conducted in this work using different images collection, to investigate the performance of our proposed method using YOLOv2.

Fine tuning YOLO-v2 achieved a satisfactory recall accuracy. Nonetheless, it detects some of the false-positive objects that yield low precision accuracy. These false-positive results will be eliminated by employing K-means clustering and their tracking optical flow information, as described in the following subsections. The false-positive results obtained from the background regions have different motion characteristics compared to the foreground vehicles. Hence we exploit their feature points motion information to discard them before the final decision. There are different clustering strategies that can be used to group the feature points as mentioned in [7, 41, 42]. However, in this work, K-means clustering is sufficient to achieve a good accuracy with a low computation complexity.



**Fig. 2** Diagram represents the ResNet-50 transfer learning architecture used for vehicle detection

### 3.2 Vehicle features refinement and clustering

This step is used to extract the vehicles and eliminate the background regions, in addition to vehicles clustering process. Optical flow-based tracking leads to a faster processing speed and feature matching accuracy. So, optical flow Kanade-Lucas algorithm [2] is used for tracking the feature points from frame  $f$  to  $f + 1$ .

The result of the optical flow in the first frame pairs is a set of vectors  $C$  with elements  $C_i = (S_i, \theta_i)$ , where  $S$  and  $\theta$  are given by

$$S_i = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2} \quad (1)$$

$$\theta_i = \arctan\left(\frac{Y_2 - Y_1}{X_2 - X_1}\right) \quad (2)$$

where  $X_1$  and  $Y_1$  represent the  $X$  and  $Y$  coordinates in the previous frame, while  $X_2$  and  $Y_2$  represent the  $X$  and  $Y$  coordinates in the next frame. Each element in  $C$  corresponds to an interest feature point  $P_i$  tracked from frame  $f$  to  $f + 1$ , where  $S$  and  $\theta$  are two vectors containing the displacement magnitudes and angles respectively for each corner point.

The noisy detections tend to result in short-lived trackers. So in this work, the foreground detection is considered a vehicle object only if it tracked in several consecutive frames (9 consecutive frames) since the noisy detections may be tracked for short-lived period. Then k-means clustering is used to cluster the remaining foreground vehicles. More details can be found in [14].

### 3.3 Vehicle counting

After extracting the most robust features and grouping them into separate clusters for each detected vehicle. Each of these vehicle features is assigned a unique ID and tracked until disappearing through the video. The assigning process is based on the intersection area of the old rectangular tracked bounding box and the new detected bounding box. If the intersection area is greater than a predetermined percentage  $\alpha$ , this vehicle has the same identity of the old matched vehicle. While there is no intersection area or having an intersection area less than  $\alpha$ , the vehicle gets a new label. There are four counting possibilities for each N-frames:

- A new vehicle detected for the first time, so all features inside its bounding box haven't assigned to a label. In this case, a new label is given to these features and the counter is increased by one.
- A vehicle bounding box is detected at the first frame and it was previously detected in the previous frameset (N-frames), thus some or all of its features have a label. In this case, the unlabeled features (if existed) take the same label.
- A vehicle bounding box is not detected in the first frame but it is detected before in the previous frameset. In this case, its features were given a label before, so the same label is continuously assigned to them.
- A vehicle is never detected through all the video framesets, this vehicle is represented as a missed counted vehicle.



**Table 1** Training experiment details

Experiment number	Source of training images	Number of training images
Experiment I	Matlab vehicle images	259
Experiment II	highway baseline scene	127
Experiment III	Matlab images, intermittenpan, street corner at night, and tramstation scenes	less than 10% of the frames of each scene

## 4 Experimental results

To investigate YOLOv2 in the detection and counting strategy under different environment, three training experiments were employed as shown in Table 1.

All experiments, including twelve videos with various challenges are used to validate the contribution of the proposed method. The Videos consist of two sequences form GRAM dataset [15], M-30 and M-30 HD, four sequences form CDnet 2014 [35], HighwayII video from ATON Testbed, and five sequences from the public urban traffic data set UA-DETRAC [36]. We test the proposed approach on nighttime, daytime, intermittent vehicle motion, and crowd scenes as mentioned in Table 2. In all experiments, the fixed number of frames in each frameset is equal to ten frames,  $N = 10$  for achieving better tracking and counting result using KLT [32].

We examined the algorithm with different values of  $\alpha$ , where we found that if  $\alpha$  is too high, the same vehicle may be classified into a new vehicle. The value of  $\alpha = 25\%$  yields to the best accuracy in our experiment.

Quantitative evaluation of the detection and counting will be discussed and compared with the state of art approach [14]. The detection accuracy is evaluated using quantitative performance metrics that have been used as a standard evaluation [18], known as Precision

**Table 2** Challenge environments information of the videos used in the performance evaluation

Dataset	Sequence	Challenging description
GRAM Dataset	M-30	Sunny day, and low resolution camera.
	M-30-HD	Cloudy day, and high resolution camera.
CDnet2014	Highway	Sunny day, shadows, and waving trees.
	Intermittenpan	Sunny day, and waving trees.
	Street corner at night	Light changes, and night scene.
	Tramstation	Night scene, and light changes.
ATON Testbed	Highway II	Crowded scene
UA-DETRAC	MVI-40852	Cloudy scene
	MVI-39401	Sunny day
	MVI-40772, MVI-40775, and MVI-40793	Night scenes



**Fig. 3** Sample Matlab training images used in the fine-tuning

and Recall. The precision is calculated as the percentage of correctly detection vehicle pixels [true positive (TP)] over the total number of detecting object pixels including TPs and false positive (FP).

$$Precision = \frac{|TP|}{|TP| + |FP|} \tag{3}$$

Recall refers to the ratio of accurately detected vehicle pixels to the number of actual vehicle pixels that include the number of false negative pixels (FN).

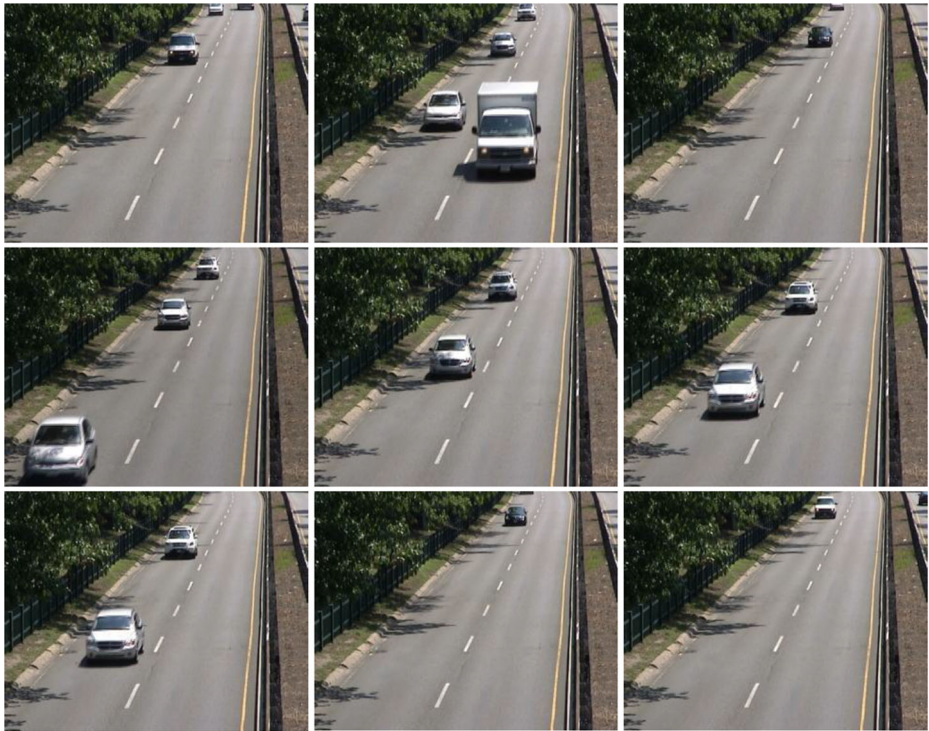
$$Recall = \frac{|TP|}{|TP| + |FN|} \tag{4}$$

The counting precision can be defined as

$$Precision (\%) = 100 - Error (\%) \tag{5}$$

Where

$$Error (\%) = \frac{|Estimated - TrueNo. |}{TrueNo.} \times 100 \tag{6}$$



**Fig. 4** Sample Highway training images used in the fine-tuning from CD-net2014 dataset

Moreover, the processing time of detection and counting is calculated to evaluate the computational time complexity of the proposed approach, represented by Frame Per Second (FPS). Time improvement is estimated according to the following equation

$$\text{Time Improvement (\%)} = \frac{|\text{Estimated FPS} - \text{Old FPS}|}{\text{Estimated FPS}} \times 100 \quad (7)$$

Where *Estimated FPS* is the number of processing frames using the proposed strategy, and *Old FPS* is the number of processing frames using the other compared strategy.

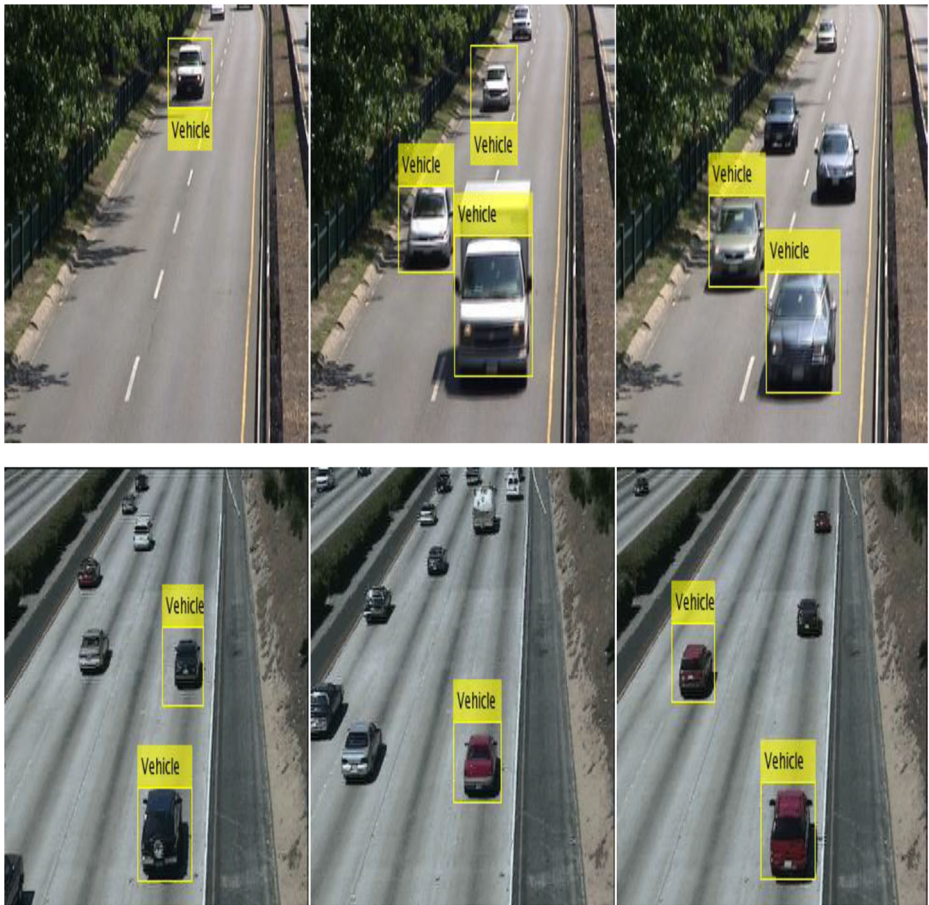
In experiment I, the Yolov2 network is re-trained on Matlab collection of vehicle images, 295 images. Samples of these images are shown in Fig. 3. In experiment II, for the sake of comparison with background subtraction based proposed method [14], 127 frames of highway baseline scene in ChangeDetection.net dataset (CDnet 2014) are used to re-train the network. Samples of these images are shown in Fig. 4.

In experiment III, other datasets were used for enhancing the accuracy, frames collection from ChangeDetection.net dataset (CDnet 2014) including, Intermittenpan, street corner at night, and tramstation videos are used for the retrain the network. Samples of these images are shown in Fig. 5. It is important to note that the image frames used for training are not used also for testing. In experiment III, we used only less than 10% of the frames of



**Fig. 5** Sample training images collected from matlab and CD-net2014 dataset sequences, used in the fine-tuning

different sequences including Tram station, Street corner at night, and Intermittenpan with the matlab images. While the remaining frames about 90% of these sequences are used in the testing. Moreover, In experiment III, we used the M-30 and M-30-HD sequences from GRAM dataset, highwayII sequence from ATON dataset, and five sequences (MVI-39401, MVI-40852, MVI-40772, MVI-40775, and MVI-40793) from UA-DETRAC dataset for testing, and these sequences are not included in the training images. The experimental results were implemented employing MATLAB on Intel i7-4810MQ CPU 2.80 GHz, 16 GB RAM, and Quadro K1100M GPU with 2 GB of video RAM. In addition, an experiment was conducted to test the performance evaluation using either CNN-based YOLOv2 for the detection or both of Yolov2 detection combined with optical flow tracking. We use the four videos of Changedetection.net 2014 (CD2014) dataset that mentioned in [14], with the same performance metrics.



**Fig. 6** Sample results show missing vehicles using experiment I

There are several pre-trained models available, but direct application to another dataset may not be feasible. The result may contain a lot of missed detections and some incorrect detections have been classified. Nonetheless, these pre-trained weights are still useful and can be used with a transfer learning approach to weight initialization and fine tuning of the network on a new dataset using a transfer learning approach.

In experiment I, the images used for training have an approximate visual feeling with the highway sequence from CDnet2014 dataset and highwayII sequence from ATON dataset, so YOLOv2, in this case can detect the vehicles from these two sequences but still the accuracy is not sufficient because some missed detections occur as shown in Fig. 6.

We evaluate the proposed methods in the two cases, the first one using background subtraction based CNN in the detection part, and the second case, using YOLOv2 based detection. As clear from Table 3, the proposed method using YOLOv2 and tracking information achieved the fastest processing speed with the lower counting precision compared to

**Table 3** Vehicle counting results comparison on CDnet2014 sequence and ATON for experiment I

Compared Algorithm			BS-based Method [14]	YOLOv2-based Proposed Method
CDnet 2014 Dataset	Highway	Miss detection	<b>0</b>	2
		Precision	<b>100</b>	91.7
		Time	0.18	<b>19.35</b>
		Performance (FPS)		
ATON Testbed	Highway II	Miss detection	<b>1</b>	33
		Precision	<b>97.9</b>	31.25
		Time	0.18	<b>29.7</b>
		Performance (FPS)		

the method based on BS-CNN in the detection phase. It is asserted that the YOLOv2 based detection increases the time performance by 99.1% and 99.4% for Highway and Highway II respectively, according to (7).

In experiment II, to improve the performance, the training images are selected from one evaluation testing scenes, 127 frames of Highway sequence from CDnet 2014 dataset. Images used for training have an approximate visual feeling with the M-30 and M-30-HD sequences from GRAM dataset and highwayII sequence from ATON dataset, so YOLOv2 in this case can detect the vehicles from these three sequences but still the accuracy is not sufficient for GRAM dataset sequences and ATON dataset sequence, because some missed detections occur as shown in Fig. 7.

We compare the proposed method in the two cases, the first one using background subtraction based CNN (BS-CNN) in the detection part, and the second case, using YOLOv2 based detection. As clear from Table 4, the method using YOLOv2 and tracking information achieved the fastest processing speed with the lower counting precision compared to background subtraction based CNN in the detection phase in CDnet2014 dataset, and ATON dataset sequences. It is asserted that the YOLOv2 based detection increases the time performance by 98.9 %, 99.3 %, 98.9 %, and 99 % for M-30, M-30-HD, Highway II, and Highway respectively. However, using the background subtraction based CNN detection case, the algorithm has the best counting precision accuracy while the same accuracy achieved when the training images from the testing scene as mentioned in Table 4 for Highway sequence.

Tables 3 and 4 show that the YOLOv2 has lower accuracy when trained on images less visually to the testing images since vehicles from two scenes using experiment one can be detected while vehicles from four scenes using experiment II can be detected. Hence in experiment III, we increased the images used for training from different scenes to improve the accuracy.

In experiment III, we used only less than 10% of the frames of different sequences including Tram station, Street corner at night, and Highway with the images of experiment I to train YOLOv2 with different visual images from the testing videos. In this case, the images used for training have an approximate visual feeling with the testing dataset, so YOLOv2

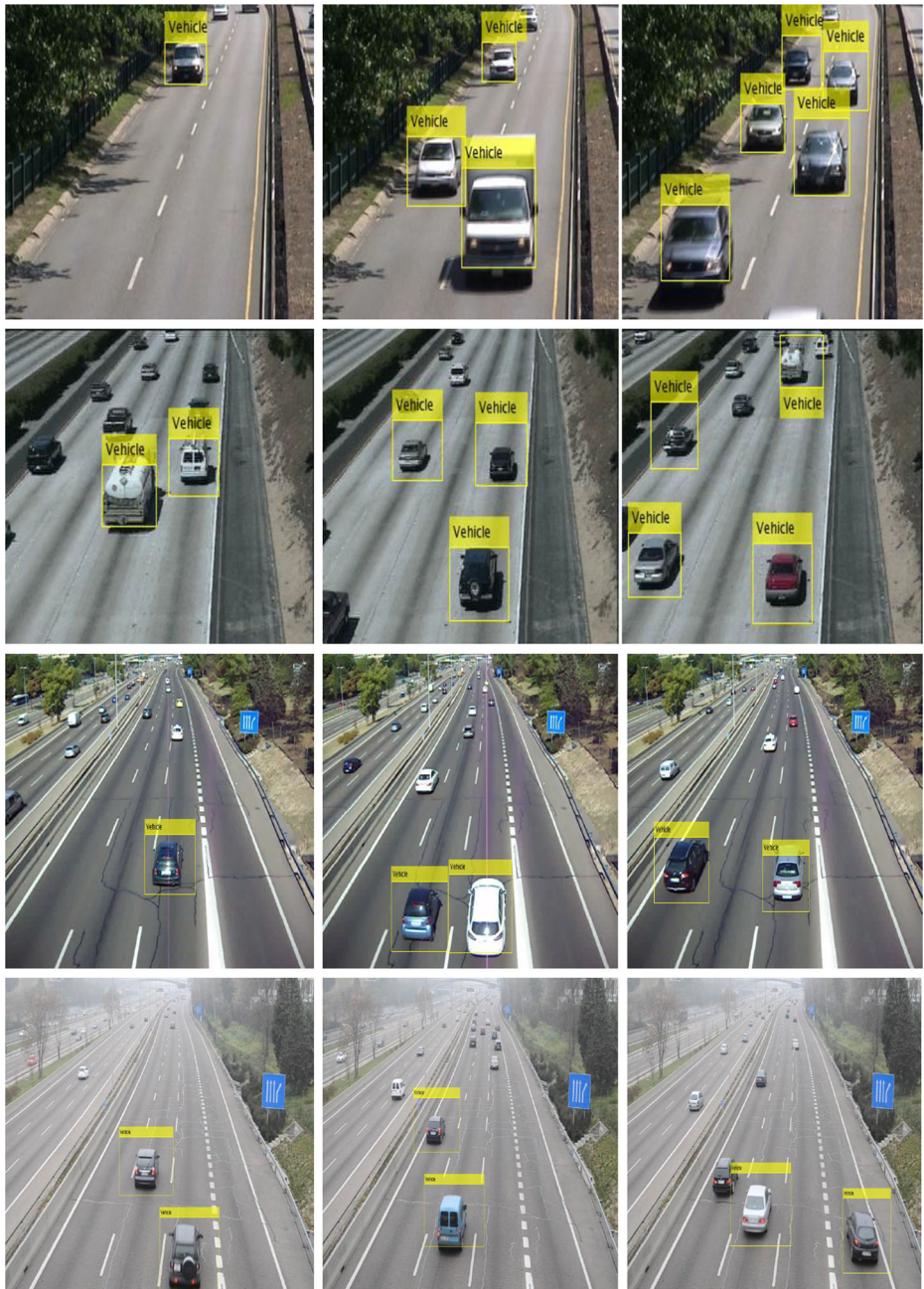


Fig. 7 Sample results show missing vehicles using experiment II

**Table 4** Vehicle counting results comparison on CDnet2014, GRAM Dataset, and ATON for experiment II.

Compared Algorithm			BS-based Method [14]	YOLOv2-based Proposed Method
GRAM Dataset	M-30	Miss detection	<b>0</b>	35
		Precision	<b>100</b>	54.6
		Time Performance (FPS)	0.225	<b>19.8</b>
	M-30-HD	Miss detection	<b>0</b>	23
		Precision	<b>100</b>	45.3
		Time Performance (FPS)	0.18	<b>26.1</b>
ATON Testbed	Highway II	Miss detection	<b>1</b>	9
		Precision	<b>97.9</b>	81.3
		Time Performance (FPS)	0.18	<b>17.55</b>
		Miss detection	<b>0</b>	<b>0</b>
CDnet 2014	Highway	Precision	<b>100</b>	<b>100</b>
		Time Performance (FPS)	0.18	<b>18.9</b>

based proposed method results in this case improved comparing to the previous experiments as shown in Figs. 8 and 9.

We compare the proposed method in three cases, the first one using background subtraction based CNN (BS-CNN) in the detection part, the second case, using faster R-CNN based detection, and in the third case using YOLOv2 based detection. In addition, we investigate the algorithm when the detection is based only on the YOLOv2 algorithm compared to the detection decision based on Yolov2 with the tracking information as shown in Table 9. As clear from Table 5, the method using YOLOv2 and tracking information achieved the fastest processing speed compared to BS-CNN and faster R-CNN used in the detection phase, for CDnet2014 dataset sequences. The evaluation experiment shows that both of faster R-CNN-based proposed method and YOLOv2-based proposed method have a comparable precision and recall results.

It is asserted that the YOLOv2 based proposed method increases the average time performance by 94.9% from 0.81 frames per second of BS-CNN case to 15.75 frames per second. However, using the BS-CNN based proposed method, the algorithm average recall accuracy outperform by 13.2%, while the average precision accuracy reduced by 3.75%. The time improvement resulted from YOLOv2 based proposed method is due to the frame testing processing without batch processing. While the accuracy improvement using BS-CNN based proposed method is due to pixel classification for all frame pixels with more time.





**Fig. 8** Sample detection results using experiment III. Top row: crowded and Cloudy. Second and fourth row: different camera view, sunny day, and waving trees. Third and Bottom row: Night scenes with changed light



**Fig. 9** Sample counting results using experiment III. Top row: crowded and Cloudy. Second and fourth row: different camera view, sunny day, and waving trees. Third and Bottom row: Night scenes with changed light

**Table 5** Vehicle detection results comparison on CDnet2014 sequences

Method		Yang et al. [37]	BS-based Method [14]	FR-CNN-based Proposed Method	YOLOv2-based Proposed Method
Highway	Precision %	91.3	98.2	<b>100</b>	<b>100</b>
	Recall %	92.1	<b>99.2</b>	95	92
	Time Performance (FPS)	12	0.18	1.31	<b>18.9</b>
Intermittentpan	Precision %	90.2	98.7	<b>100</b>	<b>100</b>
	Recall %	<b>98.8</b>	97.5	83	81
	Time Performance (FPS)	12	0.27	0.49	<b>13.95</b>
Streetcorneratnight	Precision %	89.1	95.8	<b>100</b>	<b>100</b>
	Recall %	<b>97.2</b>	97.1	78	76
	Time Performance (FPS)	12	0.225	0.63	<b>14.85</b>
Tramstation	Precision %	87.1	92.3	<b>100</b>	<b>100</b>
	Recall %	97.1	<b>97.1</b>	93	89
	Time Performance (FPS)	N/A	0.18	0.72	<b>15.3</b>
Average accuracy	Precision %	89.4	96.25	100	<b>100</b>
	Recall %	96.3	<b>97.7</b>	87.3	84.5
	Time Performance (FPS)	12	0.18	0.81	<b>15.75</b>

**Table 6** Vehicle counting results comparison on GRAM dataset and ATON Testbed

Compared Algorithm	Dataset	Yang et al. [37]	Santos et al. [30]	Li et al. [19]	BS-based Method [14]	FR-CNN-based		YOLOv2-based	
						Proposed Method	Proposed Method	Proposed Method	Proposed Method
GRAM	M-30	6	1	0	0	1	1	0	0
	Precision	92.20	98.7	100	100	98.7	98.7	100	100
	Time	12	5.67	9	0.225	0.81	0.81	22.5	22.5
M-30-HD	Performance (FPS)								
	Miss detection	5	0	0	0	0	0	0	0
	Precision	88.10	100	100	100	100	100	100	100
ATON Testbed	Highway II	3	1	N/A	1	1	1	1	1
	Precision	92.31	97.92	N/A	97.9	97.9	97.9	97.9	97.9
	Time	12	6.03	N/A	0.18	1.17	1.17	16.2	16.2
	Performance (FPS)								

**Table 7** Vehicle counting results comparison on UA-DETRAC sequences

Compared Algorithm	UA-DETRAC Dataset	Li et al. [19]	BS-based Method [14]	FR-CNN-based Proposed Method	YOLOv2-based Proposed Method
	MVI-39401	4	5	3	3
		Miss detection			
		Precision	95.45	<b>97.27</b>	<b>97.27</b>
		Time	0.12	1.4	<b>20.2</b>
		Performance (FPS)			
	MVI-40852	2	3	1	1
		Miss detection			
		Precision	94.73	<b>98.24</b>	<b>98.24</b>
		Time	0.182	0.93	<b>17.4</b>
		Performance (FPS)			
	MVI-40772	2	3	1	1
		Miss detection			
		Precision	93.6	<b>97.9</b>	<b>97.9</b>
		Time	0.21	1.4	<b>19.2</b>
		Performance (FPS)			
	MVI-40775	2	3	1	1
		Miss detection			
		Precision	92	<b>97.3</b>	<b>97.3</b>
		Time	0.2	1.29	<b>18</b>
		Performance (FPS)			
	MVI-40793	3	4	2	2
		Miss detection			
		Precision	95	<b>97.4</b>	<b>97.4</b>
		Time	0.17	1.38	<b>20</b>
		Performance (FPS)			

**Table 8** Vehicle counting results comparison on CDnet2014 sequences

Compared Algorithm	Yang et al. [37]	Santos et al. [30]	Li et al. [19]	BS-based Method [14]	FR-CNN-based Proposed Method		YOLOv2-based Proposed Method	
Highway	Precision	<b>100</b>	96.3	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>
	Time (FPS)	5.91	9	0.18	1.31	18.9	18.9	18.9
	Performance							
Intermittenpan	Precision	N/A	N/A	<b>93.3</b>	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>
	Time (FPS)	N/A	N/A	0.27	0.49	<b>13.95</b>	<b>13.95</b>	<b>13.95</b>
	Performance							
Streetcorneratnight	Precision	N/A	N/A	<b>95.2</b>	<b>95.2</b>	<b>95.2</b>	<b>95.2</b>	<b>95.2</b>
	Time (FPS)	N/A	N/A	0.225	0.63	<b>14.85</b>	<b>14.85</b>	<b>14.85</b>
	Performance							
TramStation	Precision	N/A	N/A	<b>91.6</b>	<b>91.6</b>	<b>91.6</b>	<b>91.6</b>	<b>91.6</b>
	Time (FPS)	N/A	N/A	0.18	0.72	<b>15.3</b>	<b>15.3</b>	<b>15.3</b>
	Performance							

As clear from Tables 6 and 7, the method using YOLOv2 and tracking information achieved the fastest processing speed with comparable counting precision compared to BS-CNN and Faster R-CNN in the detection phase, for GRAM Dataset, ATON Testbed, and UA-DETRAC sequences. It is asserted that the YOLOv2 based detection proposed method increases the time performance by 96.4%, 89.3%, 92.8%, 93%, 94.6%, 92.7%, 92.8%, and 93.1% for M-30, M-30-HD, Highway II, MVI-39401, MVI-40852, MVI-40772, MVI-40775, and MVI-40793, respectively, with the best counting precision accuracy. Authors in [30], proposed a new method to detect, track, and count vehicles by utilizing the detector YOLOv3 and the tracker Deep SORT (Simple Online Realtime Tracking). However, their algorithm needs more time (5.61 FPS) compared to our strategy (19.2 FPS) as shown in Table 6, because YOLOv2 with optical flow information is faster than YOLOv3 and Deep SORT tracker [23]. Table 8 present the CD net 2014 counting comparison for the proposed strategies in three cases, the first one using background subtraction based CNN in the detection part, the second case using faster R-CNN, and in the third case, using YOLOv2 based detection. The method using YOLOv2 and tracking information achieved the faster processing speed with the best counting precision with background subtraction based CNN and faster R-CNN in the detection phase in TramStation, and Intermittenpan sequences. It is asserted that the YOLOv2 based detection increases the time performance by 99%, 98.1%, 95.8, and 95.3 compared to BS-CNN based detection for Highway, Intermittenpan, Streetcorneratnight, and TramStation respectively. However, using the background subtraction based CNN detection case, the algorithm has the best counting precision accuracy in Intermittenpan, and TramStation sequences.

To investigate the influence of tracking information in our proposed algorithm, we evaluate the proposed algorithm in two cases; the first case, using only the YOLOv2 detection information without tracking information, and in the second one using both the detection and tracking information. We test the CDnet 2014 sequences using experiment III for training YOLOv2. Table 9 shows the comparison result for the two cases.

It is asserted that from Table 9, the proposed method based on collaboration between YOLOv2 and tracking information, increases the time performance by 90%, 87.7%, 88.8%, and 89.7% , 97.6%, 97.5%, 97.8% for Highway, Intermittenpan, Streetcorneratnight, TramStation, M-30, M-30-HD, and Highway II, respectively. Since the detection process takes a large time compared to the tracking process, if the detection decision was taken based only on every frame's detection without tracking information, the algorithm takes more time than the detection only once every ten frames. In the latter method, the detection in the remaining frames depends on the tracking information.

Moreover, some missed vehicles have occurred when depending only on the detection information using YOLOv2, such missed vehicles can be detected through the frames once they detected one time through the frames using the tracking information. As shown in Table 9, the YOLOv2 with tracking information improves the Recall accuracy by 7.5%, 4%, 6%, 3%, 5%, 6%, and 7% for Highway, Intermittenpan, Streetcorneratnight, TramStation, M-30, M-30-HD, and Highway II, respectively.

## 5 Conclusion

In this work, efficient vehicle detection and counting scheme based on convolution neural networks (CNNs) and KLT tracker was introduced. YOLOv2 classifier detector was used

**Table 9** Vehicle detection results comparison between YOLOv2 based only and YOLOv2 with tracking information

Method		YOLOv2	YOLOv2-based
Sequence videos		Based Decision	Proposed Method
Highway	Precision %	<b>100</b>	<b>100</b>
	Recall %	84.5	<b>92</b>
	Time Performance (FPS)	1.89	<b>18.9</b>
Intermittenpan	Precision %	<b>100</b>	<b>100</b>
	Recall %	86	<b>90</b>
	Time Performance (FPS)	1.71	<b>13.95</b>
Streetcorneratnight	Precision %	<b>100</b>	<b>100</b>
	Recall %	74	<b>80</b>
	Time Performance (FPS)	1.67	<b>14.85</b>
Tramstation	Precision %	<b>100</b>	<b>100</b>
	Recall %	87	<b>90</b>
	Time Performance (FPS)	1.57	<b>15.3</b>
M-30	Precision %	<b>100</b>	<b>100</b>
	Recall %	91	<b>96</b>
	Time Performance (FPS)	0.54	<b>22.5</b>
M-30-HD	Precision %	<b>100</b>	<b>100</b>
	Recall %	88	<b>94</b>
	Time Performance (FPS)	0.49	<b>19.8</b>
Highway II	Precision %	<b>100</b>	<b>100</b>
	Recall %	89	<b>96</b>
	Time Performance (FPS)	0.36	<b>16.2</b>
Average accuracy	Precision %	<b>100</b>	<b>100</b>
	Recall %	85.6	<b>91.1</b>
	Time Performance (FPS)	1.17	<b>17.35</b>

for vehicle detection. The detection process was performed in the first frame every frame-set (N-frames). Then, the KLT tracker had been adopted with K-means clustering to refine and cluster the detected vehicles through the N-frames. Finally, merging clusters was introduced to classify the vehicles detected with its correspondence vehicles cluster. Different experiments are conducted to investigate pixel CNN based classification and regression CNN based classification, including faster R-CNN and YOLOv2 for vehicle detection and counting. In addition, we investigate the algorithm when the detection is based only on the YOLOv2 algorithm compared to the detection decision based on YOLOv2 with the tracking information. The proposed method is faster than the background subtraction and faster R-CNN based detection. To achieve good detection and counting results using YOLOv2, the training images must include different samples of the vehicle scenes as noticed in experiment III and experiment II. Moreover, It has been presented that the tracking information improved the average Recall accuracy by 5.5% and improved the time performance by an average of 93.3% for different seven sequences. The proposed methods handled the state



of art disadvantages accompanied by traditional and CNN's approaches, with low computational complexity, better detection and counting accuracy, and working with different and complex traffic scenes.

**Funding** Open access funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB).

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Bochkovskiy A, Wang C-Y, Liao H-YM (2020) Yolov4: Optimal speed and accuracy of object detection. arXiv:2004.10934
2. Bouguet J-Y et al (2001) Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. Intel Corporation 5(1-10):4
3. Chen D-Y, Chen G-R, Wang Y-W (2013) Real-time dynamic vehicle detection on resource-limited mobile platform. IET Comput Vis 7(2):81–89
4. Chen Y, Hu W (2020) Robust vehicle detection and counting algorithm adapted to complex traffic environments with sudden illumination changes and shadows. Sensors 20(9):2686
5. Chmiel W, Dańda J, Dziech A, Ernst S, Kadłuczka P, Mikrut Z, Pawlik P, Szwed P, Wojnicki I (2016) Insignia: an intelligent transportation system for urban mobility enhancement. Multimed Tools Appl 75(17):10529–10560
6. Doulamis ND (2010) Coupled multi-object tracking and labeling for vehicle trajectory estimation and matching. Multimed Tools Appl 50(1):173–198
7. Doulamis ND, Kokkinos P, Varvarigos E (2012) Resource selection for tasks with time requirements using spectral clustering. IEEE Trans Comput 63(2):461–474
8. Farag W, Saleh Z (2019) An advanced vehicle detection and tracking scheme for selfdriving cars
9. Fu W, Zhou J, Liu S, Ma M, Ma Y (2016) Differential trajectory tracking with automatic learning of background reconstruction. Multimed Tools Appl 75(21):13001–13013
10. Girshick R (2015) Fast r-cnn. In: Proceedings of the IEEE international conference on computer vision, pp 1440–1448
11. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 580–587
12. Gomaa A, Abdelwahab MM, Abo-Zahhad M (2018) Real-time algorithm for simultaneous vehicle detection and tracking in aerial view videos. In: 2018 IEEE 61st International Midwest Symposium on Circuits and Systems (MWSCAS), IEEE, pp 222–225
13. Gomaa A, Abdelwahab MM, Abo-Zahhad M (2020) Efficient vehicle detection and tracking strategy in aerial videos by employing morphological operations and feature points motion analysis. Multimedia Tools and Applications, pp 1–21
14. Gomaa A, Abdelwahab MM, Abo-Zahhad M, Minematsu T, Taniguchi R (2019) Robust vehicle detection and counting algorithm employing a convolution neural network and optical flow. Sensors 19(20):4588
15. Guerrero-Gómez-Olmedo R, López-Sastre RJ, Maldonado-Bascón S, Fernández-Caballero A (2013) Vehicle tracking by simultaneous detection and viewpoint estimation. In: international work-conference on the interplay between natural and artificial computation, Springer, pp 306–316
16. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778

17. Karim S, Zhang Y, Yin S, Laghari AA, Brohi AA (2019) Impact of compressed and down-scaled training images on vehicle detection in remote sensing imagery. *Multimed Tools Appl* 78(22):32565–32583
18. Kasturi R, Goldgof D, Soundararajan P, Manohar V, Garofolo J, Bowers R, Boonstra M, Korzhova V, Zhang J (2008) Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol. *IEEE Trans Pattern Anal Mach Intell* 31(2):319–336
19. Li S, Chang F, Liu C, Li N (2020) Vehicle counting and traffic flow parameter estimation for dense traffic scenes. *IET Intell Transp Syst*
20. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y, Berg AC (2016) Ssd: Single shot multibox detector. In: *European conference on computer vision*, Springer, pp 21–37
21. Minematsu T, Shimada A, Uchiyama H, Taniguchi R (2018) Analytics of deep neural network-based background subtraction. *J Imaging* 4(6):78
22. Pan SJ, Yang Q (2009) A survey on transfer learning. *IEEE Trans Knowl Data Eng* 22(10):1345–1359
23. Peng G (2019) Performance and accuracy analysis in object detection
24. Quesada J, Rodriguez P (2016) Automatic vehicle counting method based on principal component pursuit background modeling. In: *2016 IEEE International Conference on Image Processing (ICIP)*, IEEE, pp 3822–3826
25. Redmon J, Farhadi A (2017) Yolo9000: better, faster, stronger. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 7263–7271
26. Redmon J, Farhadi A (2018) Yolov3: An incremental improvement. [arXiv:1804.02767](https://arxiv.org/abs/1804.02767)
27. Ren S, He K, Girshick R, Sun J (2015) Faster r-cnn: Towards real-time object detection with region proposal networks. In: *Advances in neural information processing systems*, pp 91–99
28. Rublee E, Rabaud V, Konolige K, Bradski G (2011) Orb: An efficient alternative to sift or surf. In: *2011 International conference on computer vision*. Ieee, pp 2564–2571
29. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M et al (2015) Imagenet large scale visual recognition challenge. *Int J Comput Vis* 115(3):211–252
30. Santos AM, Bastos-Filho CarmeloJA, Maciel AlexandreMA, Lima E (2020) Counting vehicle with high-precision in brazilian roads using yolov3 and deep sort. In: *2020 33rd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, IEEE, pp 69–76
31. Shakeri M, Zhang H (2016) Corola: A sequential solution to moving object detection using low-rank approximation. *Comput Vis Image Underst* 146:27–39
32. Sheorey S, Keshavamurthy S, Yu H, Nguyen H, Taylor CN (2014) Uncertainty estimation for klt tracking. In: *Asian conference on computer vision*, Springer, pp 475–487
33. Song H, Liang H, Li H, Dai Z, Yun X (2019) Vision-based vehicle detection and counting system using deep learning in highway scenes. *Eur Trans Res Rev* 11(1):51
34. Tang Y, Zhang C, Gu R, Li P, Yang B (2017) Vehicle detection and recognition for intelligent traffic surveillance system. *Multimed Tools Appl* 76(4):5817–5832
35. Wang Y, Jodoin P-M, Porikli F, Konrad J, Benezeth Y, Ishwar P (2014) C3net 2014: An expanded change detection benchmark dataset. In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp 387–394
36. Wen L, Du D, Cai Z, Lei Z, Chang M, Qi H, Lim J, Yang M, Lyu S (2015) Detrac: A new benchmark and protocol for multi-object tracking. [arXiv:1511.04136](https://arxiv.org/abs/1511.04136) 2(4):7
37. Yang H, Qu S (2017) Real-time vehicle detection and counting in complex traffic scenes using background subtraction model with low-rank decomposition. *IET Intell Transp Syst* 12(1):75–85
38. Yang Z, Pun-Cheng LSC (2018) Vehicle detection in intelligent transportation systems and its applications under varying environments: a review. *Image Vis Comput* 69:143–154
39. Zeng D, Zhu M (2018) Background subtraction using multiscale fully convolutional network. *IEEE Access* 6:16010–16021
40. Zhang Y, Zhao C, He J, Chen A (2016) Vehicles detection in complex urban traffic scenes using gaussian mixture model with confidence measurement. *IET Intell Transp Syst* 10(6):445–452
41. Zhu X, Zhang S, He W, Hu R, Lei C, Zhu P (2018) One-step multi-view spectral clustering. *IEEE Trans Knowl Data Eng* 31(10):2022–2034
42. Zhu X, Zhang S, Li Y, Zhang J, Yang L, Fang Y (2018) Low-rank sparse subspace for spectral clustering. *IEEE Trans Knowl Data Eng* 31(8):1532–1543



**Ahmed Gomaa** received the B.Sc. degree (Hons.) in electronics and communications engineering, in 2009, the M.Sc. degree from Cairo University in Electronics and Communications Engineering, Cairo, Egypt, in 2015, and the Ph.D. degree from the Electronics and Communications Engineering Department with the School of Electronics, Communications, and Computer Engineering, Egypt-Japan University of Science and Technology, in 2020. From 2009 to 2015, he was a Teaching Assistant at Electronics and Communication Department, IAET. In 2015, he joined GPS laboratory, National Research Institute of Astronomy and Geophysics (NRIAG), Egypt. In February 2019, he has joined the Laboratory for Image and Media Understanding (LIMU), Kyushu University, Japan, as a special Research Student. He is currently an Assistant Professor with the National Research Institute of Astronomy and Geophysics (NRIAG). He is a Reviewer of the IEEE Access, Signal Image and Video Processing, IET Journals, and International Journal of Remote Sensing. His current research interests include Image & Video Processing, Computer Vision, GPS,

Remote Sensing, Wireless Sensor Network, and Object Detection & Tracking.



**Tsubasa Minematsu** received the Ph.D. degree in engineering from Kyushu University, Fukuoka, Japan, in 2018. Since 2018, he has been an Assistant Professor with the Department of Advanced Information Technology Graduate School of Information Science and Electrical Engineering, Kyushu University. His research interests include image processing and pattern recognition.



**Moataz M. Abdelwahab** received the B.S. and M.S. degrees in electrical engineering from the University of Alexandria, Egypt, in 1999 and 2001, respectively, and the Ph.D. degree from the University of Central Florida, USA, in 2007. He is currently an Assistant professor in School of Electronics, Communication and Computer Engineering, Egypt- Japan University of Science and Technology (E-JUST). He is the author of more than 40 articles, one US patent. His research interests include image processing, computer vision, biomedical signal processing: face recognition, vehicle detection and tracking, and human activity recognition. Dr. Moataz was a recipient of the International Association of Pattern Recognition MVA conference Best paper award 2012 and IEEE MWSCAS 2006.



**Mohammed Abo-Zahhad** (Senior Member, IEEE, 2000) is a professor of wireless communications and multimedia processing. He received B. Sc. and M. Sc. degrees in electrical engineering, both from Assiut University (AU), Egypt in 1979 and 1983, respectively. In 1988, he received the Ph. D. degree from the University of Kent at Canterbury, UK and AU, Egypt through the channel system. He was promoted to the rank of Professor of Electronics and Communication Engineering, in Jan.1999. He has been elected the positions of vice-dean for graduate studies, from August 2006 till June 2012 and the chair of Electrical and Electronics Department from Nov. 2013 till Dec. 2016 both in the Faculty of Engineering, AU. He has been an IEEE senior member since 2000, a member of the European Society of Circuit Theory and Applications since 1998. During the period 1989-1992, he was with the Department of Telecommunications and Media Informatics, Faculty of Electrical Engineering and Informatics, Budapest University of Technology and Economics, as a research fellow. From 1996 to 2003, he joined Yarmouk University, Jordan

as a professor of electronics and communication engineering, where he taught graduate and undergraduate courses in biomedical signal processing and electronic circuits. He acted also as the General Director of the Information and Communication Technology Centers at AU and E-JUST for ten years and two years respectively. Currently, Prof. Abo-Zahhad is the Dean of School of Electronics, Communications and Computer Engineering, Egypt-Japan University of Science and Technology (E-JUST), New Borg El Arab, Alexandria-Egypt since Jan. 2017. Recently, Prof. Abo-Zahhad has been selected by the ministry of higher education as an expert for setting the study plans and regulations for new constructed Egyptian universities. He worked as a consultant for curriculum reform for Egyptian, Jordanian and Saudi Arabia Universities. He is the founder of biomedical and bioinformatics engineering undergraduate program, E-JUST. In addition, he introduced 6 graduated program in electronics and communications engineering; namely, internet of things (iot), wireless communications systems, computer vision and multimedia, wireless power transfer, embedded systems and nano-electronics and nano-photonics systems. Moreover, he introduced 5 graduated program in computer science and engineering; namely, data engineering, cybersecurity, computational neuroscience and cognitive robotics, modelling and simulation and quantum computing. His research interests include biomedical and genomic signal processing, multimedia processing, digital and switched-capacitor filters, data compression, wavelet-transforms, genetic and immune algorithms, wireless sensor nodes, massive MIMO and millimeter wave wireless communications, energy harvesting, embedded systems and internet of medical things (IoMT). He has earned many national and international research awards, among which is the Encouragement State Award in Engineering, from the Egyptian Academy of Scientific Research and Technology (ASRT), Ministry of Higher Education, Egypt, in 2005. Prof. Abo-Zahhad is also a member of the national electronics and communication promotion committee, and a accredited reviewer of the national quality assurance and accreditation authority (NAQQA), Egypt. Prof. Abo-Zahhad has published more than two hundred scientific papers in national and international conferences and impacted journals. He served as a referee for IEEE Transactions on Signal Processing, IEEE Transactions on Circuits and Systems, IEEE Access and numerous other IEEE and IET professional publications. In addition, he was the chair of the last five rounds (2017-2021) of the international Japan-Africa conference on electronics, communications and computations, organized in cooperation between E-JUST, Egypt and Kyushu University, Japan and also chaired technical sessions in many local and IEEE international conferences. He has also served as an editor of the Journal of Engineering Sciences (JES), Egypt and as an associate editor of the Journal of Engineering and Applied Science (JEAS), Springer, and the Future Internet Journal published by MDPI.



**Rin-ichiro Taniguchi** (Member, IEEE). received the B.E., M.E., and D.Eng. degrees from Kyushu University, Fukuoka, Japan, in 1978, 1980, and 1986, respectively. Since 1996, he has been a Professor with the Faculty and Graduate School of Information Science and Electrical Engineering, Kyushu University. Since 2011, he has been the Dean of the Graduate School for a period of three years. Since 2014, he has been the Director of the Research Institute for Information Technology, Kyushu University, for a period of four years. Since 2020, he has been the Vice President with Kyushu University. Since 2017, he has also been a Council Member of the Science Council of Japan. His research interests include computer vision, multimedia processing, and cyber physical systems.

## Affiliations

**Ahmed Gomaa<sup>1</sup>**  · **Tsubasa Minematsu<sup>2</sup>** · **Moataz M. Abdelwahab<sup>3</sup>** · **Mohammed Abo-Zahhad<sup>3,4</sup>** · **Rin-ichiro Taniguchi<sup>2</sup>**

Tsubasa Minematsu  
minematsu@limu.ait.kyushu-u.ac.jp

Moataz M. Abdelwahab  
moataz.abdelwahab@ejust.edu.eg

Mohammed Abo-Zahhad  
mohammed.zahhad@ejust.edu.eg

Rin-ichiro Taniguchi  
rin@kyudai.jp

<sup>1</sup> National Research Institute of Astronomy and Geophysics (NRIAG), Helwan, 11421, Egypt

<sup>2</sup> School of Information Science and Electrical Engineering, Kyushu University, 744, Motooka, Nishi-ku, Fukuoka 819-0395, Japan

<sup>3</sup> Electronics and Communications Engineering Department, Egypt-Japan University of Science and Technology, Alexandria 21934, Egypt

<sup>4</sup> Electrical and Electronics Engineering Department, Faculty of Engineering, Assiut University, Assiut 71511, Egypt