



OPEN

Smart traffic management of vehicles using faster R-CNN based deep learning method

Arindam Chaudhuri

With constant growth of civilization and modernization of cities all across the world since past few centuries smart traffic management of vehicles is one of the most sorted after problem by research community. Smart traffic management basically involves segmentation of vehicles, estimation of traffic density and tracking of vehicles. The vehicle segmentation from videos helps realization of niche applications such as monitoring of speed and estimation of traffic. When occlusions, background with clutters and traffic with density variations, this problem becomes more intractable in nature. Keeping this motivation in this research work, we investigate Faster R-CNN based deep learning method towards segmentation of vehicles. This problem is addressed in four steps viz minimization with adaptive background model, Faster R-CNN based subnet operation, Faster R-CNN initial refinement and result optimization with extended topological active nets. The computational framework uses adaptive background modeling. It also addresses shadow and illumination issues. Higher segmentation accuracy is achieved through topological active net deformable models. The topological and extended topological active nets help to achieve stated deformations. Mesh deformation is achieved with minimization of energy. The segmentation accuracy is improved with modified version of extended topological active net. The experimental results demonstrate superiority of this framework with respect to other methods.

Keywords Smart traffic management, Vehicle segmentation, Traffic density estimation, Vehicle tracking, Faster R-CNN

We are facing the challenge of continuous increase in traffic density across different cities of world. The present world population is heavily dependent on vehicles from smooth commutation purposes. This results from constant urbanization of rural areas. In order to address this problem, smart traffic management is studied by researchers as traffic regulations solution.

Human vision system has the capability to perform complex tasks very reliably and accurately. Humans detect wide spectrum of objects very easily. With recent developments in computer vision coupled with huge data sets, better algorithms and faster GPUs, precision and accuracy of object detection and classification algorithms^{1,2} have increased to an appreciable amount. For traffic monitoring vehicle localization efficiency is very important. The vehicles from Indian traffic are presented in Fig. 1. For public safety autonomous vehicle detection methods³ are in place which detect traffic objects in order to achieve correct decisions. Smart traffic management allows us to take care of various traffic related issues in an optimum manner. This is achieved using computer vision and image processing. Segmentation of vehicle acts as significant enabler in smart traffic management functionality. With occlusions, congestion as well as other environmental factors this problem becomes more severe. For smooth and effective traffic regulations, smart traffic management is always considered as low cost solution. Congestions, emergency vehicles transport, accidents and traffic related violations are easily managed through latest Artificial Intelligence algorithms. Vehicle segmentation is an important area towards smart traffic management. Some other activities here include traffic estimation, speed control and vehicle tracking. During occlusions, fog, haze, clutters and heavy traffic situations things become more complicated.

With the growth of deep learning networks vehicle detection is being studied deeply considering traffic congestion and driving safety⁴. Vehicle localization is a crucial problem⁵ in order to develop intelligent and autonomous systems. The detection of abnormalities arising from traffic violations leads to the problem of vehicle localization. This is a significant application catering the needs for variety of traffic related problems. The traffic surveillance has been a major concern in densely populated geographical areas. These days' surveillance systems are well equipped with traffic flow data where various traffic patterns are recorded. Some notable applications

Great Lakes Institute of Management, Chennai, India. email: arindamphdthesis@gmail.com



Figure 1. A real life crowded place from Indian traffic.

here include many smart cities based applications. Different versions of deep learning network based methods are in place which have provided considerable benefits for these applications³.

In this research work, we investigate Faster R-CNN based deep learning method towards segmentation of vehicles. This problem is addressed in four steps⁵: (a) minimization with adaptive background model (b) Faster R-CNN based subnet operation (c) Faster R-CNN initial refinement and (d) result optimization with extended topological active nets. Adaptive gain function is used adaptive background modeling. The gain function compensates for shadow and illumination issues. Higher accuracy in segmentation is provided by topological active net deformable model in various situations. Deformable models provide various curvatures with respect to image surfaces. The smoothness from deformation is achieved through several forces considering objects of interest. The topological and extended topological active nets are used here in order to achieve stated deformations. This helps in fitting objects on 2D surface mesh of image. The deformation of mesh is achieved with minimization of energy. For problems with complex shapes energy is changed through extended topological active net with certain thresholds. The segmentation accuracy is increased with improved version of extended topological active net. This solution has combined effect from all models using which better segmentation boundaries are achieved. The performance of stated method is compared with respect to some important metrics such as accuracy, specificity, recall, precision and F1-score values as well as Type I and Type II errors. This method provides better segmentation performance in comparison to other methods. The experimental hypothesis is justified with benchmarked datasets. The model provides appreciable results for different sets of image datasets. This research work has been approved by Research and Development Review Board at Samsung R&D Institute Delhi. This chapter is structured as follows. The literature review is presented in “[Literature review](#)” section. In “[Computational method](#)” section highlights detailed discussion on computational method. The simulation results are shown in “[Experiments and results](#)” section. Finally, conclusion is provided in “[Conclusion](#)” section.

Literature review

During recent decades’ smart traffic management involving vehicle detection has gained considerable attention. Some of the notable works are discussed here. In Ref.⁶ a pixel wise classification method based on dynamic bayesian network for vehicle detection is proposed. In Ref.⁷ an object detection scheme is presented which identifies changes in image series. Foreground vehicle segmentation using gaussian mixture model is highlighted in Ref.⁸. An adaptive background model having frames averaging with respect to time is discussed in Ref.⁹. For vehicle localization ResNet model is used in Ref.¹⁰. A vehicle classification system involving deep learning is presented in Ref.¹¹. An integrated vehicle detection and classification method is discussed in Ref.¹². Song et al.¹³ discusses YOLOv3 algorithm for vehicle detection. Lee et al.¹⁴ presents receptive field based neural network. Semantic image segmentation is used in Ref.¹⁵. Mask R-CNN with transfer learning is used in Ref.¹⁶. Shan et al.¹⁷ discusses YOLO based solution. A review on vehicle detection is highlighted in Ref.¹⁸. In Ref.¹⁹ deep learning assisted vehicle segmentation is discussed.

In Ref.²⁰ vehicles in airborne images are detected. An ensemble based method using image descriptors is discussed in Ref.²¹. In Refs.^{22,23} methods are developed through application of gaussian mixture model. In Ref.²⁴ support vector machines is used for vehicle detection. SIFT algorithm is integrated with support vector machines in Ref.²⁵ to achieve vehicle detection. For autonomous vehicles an object detection system is highlighted in Ref.²⁶. Some R-CNN version of vehicle detection algorithms are discussed in Refs.^{28,44,54}. Several significant YOLO based multi object vehicle detection algorithms are highlighted in Refs.^{27,32–35,37–39,42}. Vehicle detection algorithms in different weather conditions are presented in Refs.^{29,30,52}. Deep learning networks for vehicle detection are used in Refs.^{31,36,41,43,45,46,55,56}. Several other notable works are available in Refs.^{40,47–51,53,57}.

Computational method

In this section, smart traffic vehicle management using Faster R-CNN based deep learning based ensemble method is highlighted. The research problem revolves around traffic management which involves vehicle segmentation at different background levels⁵. In order to achieve this, we study Faster R-CNN⁵ which analyses vehicles in smart traffic. The segmentation activity on various strategic areas of smart traffic analytics provide information related to decision-making activities. All methods were performed in accordance with the relevant guidelines and regulations by Samsung R&D Institute Delhi.

Datasets used

The experimental datasets are adopted from Ref.⁵⁸. The datasets are prepared keeping in view different traffic conditions. At the very first instance multi-class vehicle objects are considered. Several challenge factors such as traffic jams and overlapping vehicles are incorporated in dataset. Broadly speaking datasets are developed with respect to two different situations viz high density and low density traffic scenes. The former comes with several objects in single image while later has single class per image having no overlapping. In order to achieve better training, images from both above mentioned situations are placed in different datasets. The high density traffic scenes are considered from several places having traffic which are less crowded in nature. A total of 1000 images from six classes of vehicles viz two wheelers, three wheelers, four wheelers, six wheelers, eight wheelers and ten wheelers are developed. Figure 2a,b highlight some sample images. High density traffic scenes are considered from daily traffic which are congested in nature. 5000 images from above stated classes are created. Certain critical factors such as illumination, occlusions etc. are incorporated irrespective of appearance, shape, scale and size in this dataset. Table 1 highlights certain high and low density dataset statistics along with annotations and class. These datasets are preprocessed and augmented as discussed in “Data annotation and augmentation” section. Each of these datasets are benchmarked with respect to standard datasets⁵⁸.

Data annotation and augmentation

In order to achieve reliable vehicle detection dataset classes are labelled. Based on motivation from Ref.⁵ an image tool⁵⁷ has been used here towards labeling and annotation of this dataset. As shown in Fig. 3a,b for each object in image, a bounding box is assigned manually. In high density datasets (HDD) from high density traffic many bounding boxes are present. In low density datasets (LDD) from low density traffic few bounding boxes are present in single image. The label of respective class is specified by these bounding boxes. As mentioned in “Datasets used” section, the whole dataset is defined through six classes. In order to increase features of datasets such that better results are obtained data preprocessing and augmentation is used. These addresses various issues which can be present in images such as noise, inconsistency and unbalanced classes. The data augmentation process used here are in lines with discussed in Refs.^{5,57}. Since augmented dataset has an unknown distribution it is benchmarked adhering to certain standards⁵⁸.

Faster R-CNN based method

Now we present a detailed description of proposed method. The method is highlighted considering four steps⁵ viz (a) minimization with adaptive background model (b) Faster R-CNN based subnet operation (c) Faster R-CNN initial refinement and (d) result optimization with extended topological active nets.



Figure 2. (a) Sample images from dataset—low density traffic (b) sample images from dataset—high density traffic.

Dataset	Low density traffic	High density traffic
Source images	5000	1400
Annotations	50,595	2000
Classes		
Two wheeler	9000	120
Three wheeler	9351	200
Four wheeler	10,879	880
Six wheeler	10,500	600
Eight wheeler	5990	110
Ten wheeler	4875	90

Table 1. Statistics of dataset images.

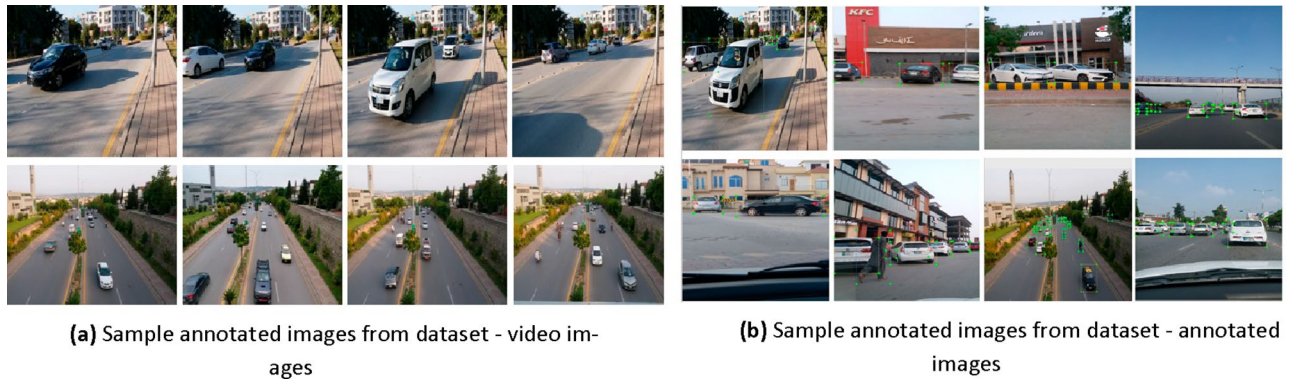


Figure 3. (a) Sample annotated images from dataset—video images (b) sample annotated images from dataset—annotated images.

Adaptive background modeling is used to construct background with traffic video as input. The video frames are analyzed in order to develop background model. The objective here is to find best background estimate. With this on foreground model shadow and illumination change impacts are minimized. This process is initiated by first initializing few frames and then continuous updates are done each time. This leads to extraction of foreground from next set of frames. The initialization of background model is done considering first frame’s pixel values. A subsequent update of model is performed through calculation of value of pixel considering background model:

$$pixel_j(y) = \overline{pixel}_{j-1}(y) + \frac{1}{GP_j \sqrt{2\pi\sigma^2}} \exp\left(\frac{-1}{2} \left(\frac{y - y_j}{\sigma}\right)^2\right) \quad (1)$$

Here y_j represents value of pixel with respect to j th frame. The value of \overline{pixel}_j is computed as follows:

$$\overline{pixel}_{j-1}(y) = \frac{pixel_j(y)}{\sum_j pixel_j(y)} \quad (2)$$

Here number of frames considered is N . The gain parameter is denoted by GP_j . It takes care of background modeling’s learning rate. The new information is learned by increasing GP_j as presented in Eq. (3) with prior information disappearing slowly.

$$GP_j = Gain \left(\frac{2}{1 + \exp\left(-\frac{cont - \alpha}{\beta}\right)} \right) \quad (3)$$

Here $Gain$ and α parameters take care of sinusoid function’s inflection point. The parameter β controls gradient. The parameter $cont$ depends on number of frames. Background model is updated for every frame. Considering every frame with background as adaptive, after its subtraction we reach to foreground objects. After adaptive background model based minimization is performed, we present basic Faster R-CNN architecture used in this research. The architecture of Faster R-CNN is considered from Ref.⁵ having variation based baseline considered from Ref.⁴⁵. The architecture is highlighted in Fig. 4. In order to validate proposed method, datasets mentioned in “Datasets used” section are used.

The convolutional feature map is developed when entire image is processed convolutional and max pooling layers. For each object’s ROI pooling layer, feature vector of fixed length is extracted. The input of sequence of

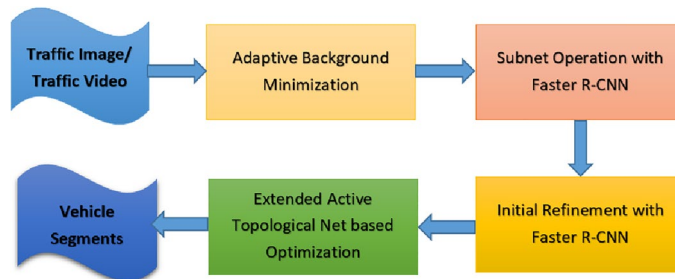


Figure 4. Architecture of vehicle segmentation method.

fully connected layers are feature vectors. From a set of fully connected layers, output passes to sibling layers where softmax probability estimates are produced. The object classes estimate softmax probability. Here background class with set of other layers are considered. The encoding is performed on set of values with respect to refined positions in bounding box for each object class. For RoI pooling layer we use max pooling. This allows conversion of features into small feature map. Here convolutional feature map based RoI is defined through 4-tuple. RoI max pooling divides rectangular window into sub window grids. The channel independent pooling is applied for each feature map. The network weights are trained through backpropagation in Faster R-CNN. A hierarchical sampling of RoIs is performed for each image. Stochastic gradient descent (SGD) training for Faster R-CNN is done in mini-batches. For training with larger datasets, execution of SGD is done for more iterations. Faster R-CNN makes use of sibling output layers. For each RoI we have discrete probability distribution as initial output. The probabilities for fully connected layer have outputs as softmax. The labeling for RoI is performed with respect to ground truth class. For each training bounding box regression having ground truth is considered. For each labeled RoI considering multitask loss, joint classification is present with respect to training and bounding-box regression. The optimization of multitask loss is performed as highlighted in Ref.⁵. For whole image classification, convolutional layers' calculation time is greater than fully connected layers. RoIs processing time is appreciably large for detection.

Now subnet operation with Faster R-CNN is presented. The foreground image considers a mesh. This is moved towards subnet operation for Faster R-CNN. The training image dataset is adopted from Ref.¹⁹. This is marked with respect to active net grids. The subnet is represented as binary matrix with 1 as indicator for link presence of mesh and 0 as indicator for link absence. Here Faster R-CNN is trained where features are learned in order to produce ground truth specific results. The U-Net based CNN Ref.⁵⁹ helps us to perform initial subnet operation. The extended topological active net refines initial subnet. Faster R-CNN produces hole based mesh considering all background. The objects have mesh nodes. Those mesh nodes which where boundary is not crossed are removed. It is reached though extended topological active net energy minimization. The energy function used here is highlighted in Eq. (4). By using Eq. (1) adaptive background model is developed.

$$Energy(w(a, b)) = \int_0^1 \int_0^1 [Energy_{internal}(w(a, b)) + Energy_{external}(w(a, b))] dadb \quad (4)$$

The mesh deformation is done using greedy approach. In situations with complex deformation of shapes, extended topological active net produces different energy term. This it achieves through various thresholds. When clutters and occlusion are present in vehicle image segmentation extended topological active net always reaches local minima. This results in low segmentation accuracy. The improved version of extended topological active net provides resolution to these problems. The combined effect of all models used here leads us to better results.

Training process

The training process is now briefly discussed. The annotated and augmented data is trained using Faster R-CNN algorithm. In order to perform training, several parameters like size of batch, epochs needed and resolution of image are used. Since network is trained from scratch, random weight initialization is performed. Here initially trained COCO weights⁵ are used towards model training with appreciable time and computation benefits. The best weight values are obtained using initially trained Faster R-CNN having transfer learning. Chaudhuri⁵ datasets are used as benchmark in order to train stated custom datasets. The batch sizes of 5, 10, 20, 30 and 40 are considered. The epochs are also changed to 100, 200, 300, 400 and 500. The confidence values are considered between 0.4 and 0.6. The best weights are used to detect objects in datasets. The predicted labels and assessment images incorporating bounding boxes with confidence values are also obtained.

Evaluation criteria

Some of the significant metrics in evaluation of smart cities include⁵ key process indicators alignment with respect to several community priorities spanning neighborhoods, alignment of investment with respect to community priorities, investment efficiency, information flow density, infrastructure services and community benefits inherent quality. In this research various evaluation metrics⁵ are used for measuring performance of our method which are discussed in “Experiments and results” section. These metrics help to identify efficiency and robustness of proposed method. Here, mean average precision (mAP) is calculated for recall values lying between 0 and 1. Along with this some comparative analysis of results is also performed.

Ethical approval and informed consent

Author has ethical and informed consent for data used in this research.

Experiments and results

Here a detailed discussion on simulation results is presented. We conducted detailed experiments in Google Colab having T4 GPU with Intel Xeon CPU and 64 GB of RAM. Python version 3.11.5 has been used as simulation tool in this research. In order to assess vehicle detection method performance, several state-of-the-art detectors are evaluated. Also various comparisons are performed with stated method considering accuracy and execution times. All methods are trained on data adapted from COCO⁶⁰ and DAWN⁶¹ datasets.

The detection and segmentation of objects is performed by COCO dataset considering natural contexts⁵. In Table 2 COCO dataset highlights several objects collected from complex scenes. The dataset has images of 100 different object types with 3 million instance labels. The results are highlighted in Fig. 5. Certain YOLOv5 semantics are adopted from Ref.⁵⁷. All vehicles are accurately detected by stated method considering variation

Dataset	Number of classes	Training	Validation	Testing
HDD	6	16,799	20,575	20,685
LDD	6	80,577	10,879	10,780
COCO	90	300,000	18,000	12,000

Table 2. Certain dataset specific statistics. *HDD* high density dataset from high density traffic, *LDD* low density dataset from low density traffic.



Figure 5. COCO dataset vehicle detection results.

Computation backbone	Input size	Multi-scale	mAP (%)
CSPDarkNet ⁵³	512 × 512	False	48.64
CNN ⁴³	512 × 512	False	49.00
R-CNN ⁴⁴	512 × 512	False	47.40
BottleneckCSP ⁴⁵	512 × 512	False	28.99
VGGNet-16 ⁴⁷	512 × 512	False	30.40
ResNet-101-FPN ⁴⁸	512 × 512	False	40.40
VGGNet-16 ⁴⁹	800 × 800	False	42.00
ResNet-101 ⁵⁰	800 × 800	False	49.40
ResNet-101 ⁵¹	512 × 512	False	40.50
CNN + SVM ⁵²	512 × 512	False	50.10
BN + ReLU ⁵³	512 × 512	False	34.90
ResNet-C4-FPN ⁵⁴	512 × 512	False	32.88
ResNet-50 ⁵⁵	512 × 512	False	50.90
SiNet ⁵⁶	512 × 512	False	51.50
CSF ⁵⁷	512 × 512	False	52.45
Our method	512 × 512	False	58.90

Table 3. COCO dataset accuracy comparison.

in illumination. In order to further validate superiority of stated method we perform a comparative analysis with 14 methods as presented in Table 3. Some significant observations are briefly highlighted here. For images with different resolutions, stated computational structure provides best performance in terms of mAP values. The methods presented in Refs.^{49,50,55,56} also provide promising results with respect to COCO datasets. He et al.⁵⁴ makes use of ResNet and produces results on lower side for object detection in COCO datasets. Similar results are obtained from methods presented in Refs.^{45,47}. Here miscellaneous objects are detected on COCO datasets with promising results. These include vehicles of varying shapes and sizes. All results are highlighted in Fig. 5.

The DAWN dataset is used in order to study and investigate performance of stated method. DAWN dataset has 2000 image with different variations⁵. It shows varying traffic situations in different weather conditions. Figure 6 shows fairly detailed results considering significant observations. The stated method addresses all prevailing weather situations such as rainy days, normal dry days and snowy days. A comparison is performed with results presented in Table 4. Some observations are highlighted here. In fog situation⁵⁴ has highest rank. Our method exceeds results of Ref.⁵⁴. Here Refs.^{43,52} have lowest ranks. In rain situation our method produces best results followed by Refs.^{45,48,50,51,57}. Here Ref.⁵² has lowest rank. In snow situation our method has highest rank followed by Refs.^{51,57}. Here Refs.^{45,47,50,51} produce similar results. In dry day situation stated method achieves highest rank. Here Refs.^{42,48,49} yield similar results. Min et al.⁵² has lowest rank.

Now we present some additional insights in this research with respect to diverse range of environments. All state-of-the-art object detection methods have been studied in this research. These results are briefly discussed here. In Ref.⁴² BIT-Vehicle and UADETRAC datasets are investigated. In Ref.⁵⁷ 3 different datasets are studied which several variations with respect to road conditions, weather as well as complex background. In Refs.^{43,48}



Figure 6. Vehicle detection results on DAWN datasets.

Computation backbone	Fog	Rail	Snow	Dry day
CSPDarkNet53 ⁴²	26.40	31.55	39.95	24.10
CNN ⁴³	24.00	21.10	38.32	23.80
R-CNN ⁴⁴	27.20	21.30	28.30	18.00
BottleneckCSP ⁴⁵	29.31	41.21	43.00	24.02
VGGNet-16 ⁴⁷	23.40	24.60	37.90	15.83
ResNet-101-FPN ⁴⁸	28.95	41.10	43.00	24.09
VGGNet-16 ⁴⁹	23.10	27.65	34.00	24.10
ResNet-101 ⁵⁰	29.70	40.10	43.00	23.99
resnet-101 ⁵¹	28.10	40.40	43.02	24.10
CNN + SVM ⁵²	16.50	14.08	15.38	10.69
BN + ReLU ⁵³	25.08	19.14	23.18	17.38
FtesNet-C4-FPN ⁵⁴	29.68	30.32	33.93	24.00
ResNet-50 ⁵⁵	28.83	27.68	30.19	24.03
SiNet ⁵⁶	26.45	20.09	27.92	11.31
CSP ⁵⁷	29.66	41.21	43.01	24.14
Our method	30.40	45.55	45.79	25.60

Table 4. Comparison of methods on DAWN datasets.

primary concentrations are on KITTI and DAWN datasets with certain deviations which produces appreciable results. In Ref.⁴⁴ investigation of object detection capability is performed on COCO and DAWN datasets. In Ref.⁴⁵ CARLA dataset is explored. The results presented here extends method's detection capability to 3 other datasets. PASCAL VOC 2007 dataset is studied in Refs.^{47,51}. In Ref.³⁷ PASCAL dataset is studied which contains annotated images of various objects. Here detection capability of method is expanded to 3 different vehicle datasets. In Refs.^{49,50,53,54} COCO and DAWN datasets are used in order to validate methods for object detection. In Ref.⁵⁴ various object detection methods are presented for detection of different objects. PETS2009 and changedetection.net 2012 datasets are studied in Ref.⁵² with DLR Munich vehicle and VEDAI datasets in Ref.⁵⁵ and KITTI and LSVH datasets in Ref.⁵⁶. The vehicle detection methods on these datasets produce appreciable results. In DAWN dataset has more challenging images with fog, rain, dry day and snow. However, as shown in Fig. 7 very low vehicle detection results are achieved. But our method performs well in this situation. Considering several state-of-the-art approaches, a detailed comparison of is performed alongwith our method. It is seen that in different situations our method performs well. In COCO dataset our method produces appreciable results in comparison to other methods.

The metrics for Type I and Type II errors account for false positives and false negatives respectively. The Table 5 highlight results with respect to Type I and Type II errors for HDD, LDD, COCO and DAWN datasets. The illustrative loss functions are applied to Type I and Type II errors. A loss function is imposed when accuracy report is chosen. When loss function is minimized accuracy increases. As a result of this care needs to be enforced when using loss functions. Here loss is reduced through minimization of objective function⁵ which is weighted sum of localization and confidence losses⁵. The localization loss is smooth L1 loss between true values and predicted bounding box correction. The coordinate correction transformation is identical to R-CNN in bounding box regression. The confidence loss represents how likely an object is contained in bounding box. It is calculated using logistic regression function based on intersection over union (IoU) between predicted bounding box and ground truth bounding box. These results are further strengthened with accuracy, specificity, sensitivity (recall), precision and F1-score values in Table 6. In order to highlight significance of results more, Fig. 8 represents comparative performance of our method with respect to structural similarity index, spatial overlap distance and hausdroff distance^{5,15,16,21} metrics.



Figure 7. Vehicle detection method on sample image.

Dataset	Classes	Validation	Errors in validation		Testing	Errors in testing	
			Type I	Type II		Type I	Type II
HDD	6	20575	277	200	20,685	335	236
LDD	6	10,879	176	160	10,780	184	175
COCO	90	18,000	265	180	12,000	207	205
DAWN	4	12,879	187	179	12,790	189	177

Table 5. Type I and type II errors for HDD, LDD, COCO and DAWN datasets.

Dataset	Performance metrics				
	Accuracy	Specificity	Sensitivity	Precision	F1-Score
HDD	0.9724	0.9770	0.9679	0.9678	0.9725
LDD	0.9666	0.9662	0.9672	0.9670	0.9679
COCO	0.9656	0.9646	0.9666	0.9667	0.9668
DAWN	0.9714	0.9713	0.9714	0.9715	0.9723

Table 6. Achieved results of our method with accuracy, specificity, sensitivity, precision and F1-score performance metrics for HDD, LDD, COCO and DAWN datasets.

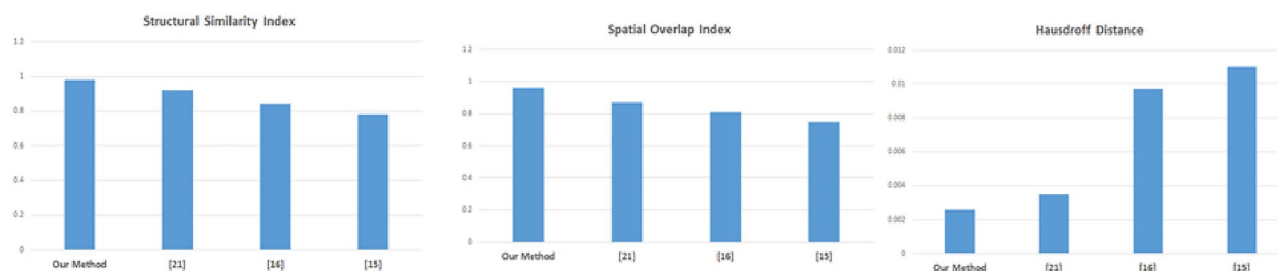


Figure 8. Comparative performance of our method considering structural similarity index, spatial overlap distance and hausdroff distance.

Conclusion

In this research work smart traffic management of vehicles is studied using Faster R-CNN based deep learning method. It is an intractable problem in computer vision and artificial intelligence domain. When occlusions, background with clutters and traffic with density variations are present, this problem becomes more challenging. The computational paradigm involves four steps viz minimization with adaptive background model, Faster R-CNN based subnet operation, Faster R-CNN initial refinement and result optimization with extended topological active nets. The concept of adaptive background modeling is incorporated in this framework. The shadow and illumination related issues are also addressed. The topological active net deformable models help to achieve higher segmentation accuracy. The deformations are reached with topological and extended topological active nets. Mesh deformation helps in minimization of energy. The segmentation accuracy is improved with modified version of extended topological active net. The superiority of this method in comparison to other methods is highlighted with experimental results. This achieved using different performance metrics such as Type I (for false positives) and Type II (for false negatives) errors, accuracy, specificity, sensitivity (recall), precision and F1-score values. The results are made more significant through comparative performance of our method with other methods with respect to structural similarity index, spatial overlap distance and hausdroff distance metrics.

Data availability

The datasets used and/or analysed during the current study available from the corresponding author on reasonable request. All data used in this research has been developed at Samsung R & D Institute New Delhi India as mentioned in Reference 58. The Institution does not allow to share and provide access to research data to public domains. In view of this, data used in this research cannot be shared. However, two more datasets COCO and DAWN used in this research are highlighted in References 60 and 61 can be shared and accessed.

Received: 11 January 2024; Accepted: 25 April 2024

Published online: 06 May 2024

References

- Mahmood, *et al.* Towards a fully automated car parking system. *IET Int. Trans. Syst.* **13**, 293–302 (2018).
- Xiaohong, *et al.* Real time object detection based on YOLOv2 for tiny vehicle objects. *Proc. Comp. Sci.* **183**, 61–72 (2021).
- Jamiya, *et al.* Little YOLO-SPP: A delicate real time vehicle detection algorithm. *Optik* **225**, 165818 (2021).
- Tajar, *et al.* A lightweight Tiny-YOLOv3 vehicle detection approach. *J. Real Time Image Proc.* **18**, 2389–2401 (2021).
- Chaudhuri, A. Smart traffic management in varying weather conditions. *Tech. Rep.*, Samsung R & D Inst. New Delhi, India (2018).
- Cheng, *et al.* Vehicle detection in aerial surveillance using dynamic bayesian networks. *IEEE Trans. Image Proc.* **21**(4), 2152–2159 (2012).
- McHugh, *et al.* Foreground adaptive background subtraction. *IEEE Sig. Proc. Lett.* **16**, 390–393 (2009).
- Chen, *et al.* *Road vehicle classification using support vector machines* (IEEE Press, 2009).
- Kul, *et al.* Distributed and collaborative real time vehicle detection and classification over video streams. *Int. J. Adv. Robot. Sys.* **14**(4), 1729881417720782 (2017).
- Jung *et al.* ResNet based vehicle classification and localization in traffic surveillance systems. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 934–940 (IEEE Press, Honolulu, 2017).
- Maungmai *et al.* Vehicle classification with deep learning. In *Proc. IEEE International Conference on Computer and Communication Systems (ICCCS)*, 294–298 (IEEE Press, Singapore, 2019).

12. Bai, *et al.* Classify vehicles in traffic scene images with deformable part based models. *Mach. Vis. App.* **29**, 393–403 (2017).
13. Song, *et al.* Vision based vehicle detection and counting system using deep learning in highway scenes. *Euro. Trans. Res. Rev.* **11**, 51 (2019).
14. Lee, *et al.* Segmentation of vehicles and roads by low channel lidar. *IEEE Trans. Int. Transp. Sys.* **20**(11), 4251–4256 (2019).
15. Kaymak *et al.* Semantic image segmentation for autonomous driving using fully convolutional networks. In *Proc. International Artificial Intelligence and Data Processing Symposium* (2019).
16. Ojha *et al.* Vehicle detection through instance segmentation using mask R-CNN for intelligent vehicle system. In *Proc. International Conference on Intelligent Computing and Control Systems (ICICCS)*, 954–959 (2021).
17. Shan *et al.* A real time vehicle detection algorithm based on instance segmentation. In *Proc. IEEE 2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, 239–243 (IEEE Press, 2021).
18. Shobha *et al.* A Review on video based vehicle detection, recognition and tracking. In *Proc. 2018 3rd International conference on computational systems and information technology for sustainable solutions (CSITSS)* (IEEE Press, 2018).
19. Shobha, *et al.* Deep learning assisted active net segmentation of vehicles for smart traffic management. *Proc. Glob. Trans.* **2**, 282–286 (2021).
20. Hamsa *et al.* Automatic vehicle detection from aerial images using cascaded support vector machine and Gaussian mixture model. In *Proc. 2018 International Conference on Signal Processing and Information Security (ICSPIS)*, 1–4 (Dubai, 2018)
21. Mikaty, *et al.* Detection of cars in high resolution aerial images of complex urban environments. *IEEE Trans. Geosci. Remote Sens.* **55**, 5913–5924 (2017).
22. Ari, *et al.* Detection of compound structures using Gaussian mixture model with spectral and spatial constraints. *IEEE Trans. Geosci. Remote Sens.* **52**, 6627–6638 (2014).
23. Hbaieb *et al.* Pedestrian detection for autonomous driving within cooperative communication system. In *IEEE Wireless Communication Network Conference*, 1–6 (IEEE Press, Morocco, 2019).
24. Xiong *et al.* High speed front vehicle detection based on video multi-feature fusion. In *Proc. 2020 IEEE 10th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, 348–351 (IEEE Press, Beijing, 2020).
25. Yawen *et al.* Research on vehicle detection technology based on SIFT features. In *Proc. 8th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, 274–278 (Beijing, 2018).
26. Li, *et al.* A deep learning based hybrid framework for object detection and recognition in autonomous driving. *IEEE Access* **8**, 194228–194239 (2020).
27. Li, *et al.* YOLO-ACN: Focusing on small target and occluded object detection. *IEEE Access* **8**, 227288–227303 (2020).
28. Cai *et al.* Cascade R-CNN: Delving into high quality object detection. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 6154–6162 (IEEE Press, USA, 2018).
29. Wang *et al.* A high precision fast smoky vehicle detection method based on improved YOLOv5 Network. In *Proc. 2021 IEEE International Conference on Artificial Intelligence and Industrial Design (AIID)*, 255–259 (IEEE Press, Guangzhou, 2021).
30. Miao *et al.* A nighttime vehicle detection method based on YOLOv3. In *Proc. Chinese Auto. Cong.*, Shanghai, 6617–6621 (2020).
31. Sarda *et al.* Object detection for autonomous driving using YOLO algorithm. In *Proc. 3rd Int. Conf. Int. Comm. Tech. Vir. Mob. Net.*, Tirunelveli, 1370–1374 (2021).
32. Zhao *et al.* Vehicle detection based on improved YOLOv3 algorithm. In *Proc. 2020 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS)*, 76–79 (Vientiane, 2020).
33. Corović *et al.* The real time detection of traffic participants using YOLO algorithm. In *Proc. 26th Telecommunication Forum*, 1–4 (Belgrade, Serbia, 2018).
34. Lou *et al.* Vehicle detection of traffic flow video using deep learning. In *Proc. 2019 IEEE 8th Data Driven Control and Learning Systems Conference (DDCLS)*, 1012–1017 (IEEE Press, Dali, China, 2019).
35. Machiraju *et al.* Object detection and tracking for community surveillance using transfer learning. In *Proc. 6th International Conference on Inventive Computation Technologies (ICICT)*, 1035–1042 (Coimbatore, India, 2021).
36. Snegireva *et al.* Vehicle classification application on video using YOLOv5 architecture. In *Proc. 2021 International Russian Automation Conference (RusAutoCon)*, 1008–1013 (Sochi, Russia, 2021)
37. Jana *et al.* YOLO based detection and classification of objects in video records. In *Proc. 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, 2448–2452 (IEEE Press, India, 2018).
38. Hu *et al.* A video streaming vehicle detection algorithm based on YOLOv4. In *Proc. IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, 2081–2086 (IEEE Press, Chongqing, China, 2021).
39. Kasper, *et al.* Detecting heavy goods vehicles in rest areas in winter conditions using YOLOv5. *Algorithms* **14**, 114 (2021).
40. Carvalho, De. *et al.* Bounding box free instance segmentation using semi-supervised iterative learning for vehicle detection. *IEEE J. Sel. Top. Appl. Earth Obsv. Remote Sens.* **15**, 3403–3420 (2022).
41. Tayara, *et al.* Vehicle detection and counting in high-resolution aerial images using convolutional regression neural network. *IEEE Access* **6**, 2220–2230 (2018).
42. Zhao, *et al.* Improved vision based vehicle detection and classification by optimized YOLOv4. *IEEE Access* **10**, 8590–8603 (2022).
43. Hassaballah, *et al.* Vehicle detection and tracking in adverse weather using deep learning framework. *IEEE Trans. Int. Transp. Syst.* **22**, 4230–4242 (2022).
44. Mahmood, *et al.* Towards automatic license plate detection. *Sensors* **22**, 1245 (2022).
45. Wu *et al.* Real time vehicle and distance detection based on improved YOLOv5 network. In *Proc. 3rd World Symposium Artificial Intelligence*, 24–28 (Guangzhou, China, 2021).
46. Lin *et al.* Microsoft COCO: Common objects in context. In *Proc. 13th European Conference on Computer Vision*, 740–755 (Zurich, Switzerland, 2014).
47. He *et al.* Bounding box regression with uncertainty for accurate object detection. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2888–2897 (IEEE Press, Long Beach, CA, USA, 2019).
48. Liu, *et al.* Dynamic vehicle detection with sparse point clouds based on PE-CPD. *IEEE Trans. Int. Transp. Syst.* **20**, 1964–1977 (2019).
49. Zhao, *et al.* M2Det: A single shot object detector based on multi-level feature pyramid network. *AAAI Conf. Art. Int.* **33**, 9259–9266 (2019).
50. Li *et al.* Scale aware trident networks for object detection. [arXiv:1901.01892](https://arxiv.org/abs/1901.01892) (2019).
51. Zhang, *et al.* Mask SSD: An effective single-stage approach to object instance segmentation. *IEEE Tran. Image Proc.* **29**, 2078–2093 (2020).
52. Min, *et al.* A new approach to track multiple vehicles with the combination of robust detection and two classifiers. *IEEE Trans. Int. Transp. Syst.* **19**, 174–186 (2018).
53. Law *et al.* CornerNet: Detecting objects as paired key points. In *European Conference on Computer Vision*, 734–750 (Glasgow, UK, 2018)
54. He *et al.* Mask R-CNN. In *Proc. IEEE International Conference on Computer Vision*, 2961–2969 (IEEE Press, Venice, Italy, 2017).
55. Shi, *et al.* Orientation aware vehicle detection in aerial images via an anchor free object detection approach. *IEEE Trans. Geosci. Remote Sens.* **59**, 5221–5233 (2021).
56. Hu, *et al.* S1Net: A scale insensitive convolutional neural network for fast vehicle detection. *IEEE Trans. Int. Transp. Syst.* **20**, 1010–1019 (2019).

57. Farid, *et al.* A fast and accurate real time vehicle detection method using deep learning for unconstrained environments. *App. Sci.* **13**, 3059 (2023).
58. Chaudhuri, A. Benchmarked datasets for vehicle segmentation. In *Technical Report* (Samsung R & D Inst., New Delhi, India 2018).
59. Ronneberger, *et al.* U-Net: Convolutional net- works for biomedical image segmentation. *Lect. Notes Comput. Sci.* **9351**, 234–241 (2015).
60. COCO Datasets: <https://cocodataset.org/#home>
61. DAWN Datasets: <https://paperswithcode.com/dataset/dawn>

Author contributions

Author contributed to the study conception and design. Material preparation, data collection and analysis were performed by A.C. The first draft of manuscript was written by A.C. and author commented on previous versions of manuscript. Author read and approved the final manuscript.

Competing interests

The author declares no competing interests.

Additional information

Correspondence and requests for materials should be addressed to A.C.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



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/>.

© The Author(s) 2024