AVE Trends in Intelligent Computing Systems

Real-time Identification of Traffic Violations through Deep Learning Techniques

O. Jeba Singh[1,*], S. Rubin Bose[2], J. Angelin Jeba[3], B. Judy Flavia[4], M. Rehena Sulthana[5]

[1]Centre for Academic Research, Alliance University, Bengaluru, Karnataka, India.
[2]Department of Electronics and Communication Engineering, SRM Institute of Science and Technology, Ramapuram, Chennai, Tamil Nadu, India.
[3]Department of Electronics and Communication Engineering, S.A. Engineering College, Chennai, Tamil Nadu, India.
[4] Department of Computer Science and Engineering, SRM Institute of Science and Technology, Ramapuram, Chennai, Tamil Nadu, India.
[5]School of Information Technology and Engineering, Melbourne Institute of Technology, Melbourne, Victoria, Australia.
jeba.singh@alliance.edu.in[1], rubinbos@srmist.edu.in[2], angelinjeba@saec.ac.in[3], judyflav@srmist.edu.in[4], rsulthana@academic.mit.edu.au[5]

*Corresponding author

Abstract: In the majority of developing countries, the traffic management system has emerged as a significant problem in the present day. Daily, the number of vehicles continues to rise, and with it, the number of infractions of traffic rules. It is necessary to have an automated surveillance system to efficiently monitor infractions of traffic rules. The purpose of this research is to present a method for the automatic detection of visual traffic offences by utilising a Deep Learning algorithm. In this study, moving cars that violate traffic rules are considered. Some examples of these violations include driving without a helmet and crossing a signal when the lights are red. In light of this, the vehicles that violate traffic rules are identified by the use of deep learning algorithms in three stages: (1) the detection of vehicle objects, (2) the identification of traffic violations, and (3) the recognition of the license plate of the vehicle that has violated the traffic rules. An average accuracy of 95.8% was found in the detection of vehicle objects, according to the results of the tests conducted using the public data set for Indian traffic signals. Concurrently, the method was able to obtain a precision of 98.5% in the detection of violations and 93.67% in the recognition of identification plates.

Keywords: Traffic Violation; Deep Learning; Traffic Management; Automated Detection, Region Proposal Network; Fully Convolutional Neural Network, Automatic License Plate Recognition; Radio frequency Identification.

Cite as: O. J. Singh, S. R. Bose, J. A. Jeba, B. J. Flavia, and M. R. Sulthana, "Real-time Identification of Traffic Violations through Deep Learning Techniques," *AVE Trends in Intelligent Computing Systems*, vol. 2, no. 2, pp. 99–110, 2025.

Journal Homepage: https://www.avepubs.com/user/journals/details/ATICS

Received on: 29/09/2024, **Revised on:** 20/12/2024, **Accepted on:** 04/02/2025, **Published on:** 07/06/2025

DOI: https://doi.org/10.64091/ATICS.2025.000134

1. Introduction

Due to the growing population, the number of vehicles is also increasing tremendously. The rapid growth of both light and heavy vehicles on the road leads to increased traffic congestion and a higher incidence of accidents. The World Health Organisation (WHO) surveyed in 2013, and the report says that around 1.24 million people are killed due to roadside accidents

[1]. In this sense, road and driving safety is a global problem that is to be seriously considered. To prevent such accidents, traffic rules must be strictly enforced [5]. Modern technology can only find a solution to monitor vehicles that violate traffic rules. Therefore, a traffic management system is much needed to overcome the issues related to road safety [8]; [9].

In developed countries, a modern traffic management system based on information and communication technology has been installed [2]. In India, there are numerous challenges in tracing vehicles due to the variety of fonts and custom-designed plates [4]. Each vehicle has a different format and style of plate [14]. In addition, another problem that persists on Indian roads is the vast variety of vehicles from different brands and models [6]. Typically, two categories of vehicle detection systems are employed: hardware-based detection and software-based detection [7]. In a few hardware detection systems, a laser beam and a photodiode are used for vehicle detection. The laser detector can operate for 24 hours under various climatic conditions, but its performance is reduced when operated at high temperatures. Hence, optical detectors are used as two light-activated optical sensors for vehicle detection [5]. To mitigate the adverse effects of sensors in hardware-based detection, image processing techniques have been recently implemented. In this aspect, software-based detection systems are designed using complex image processing techniques [12]. The stages, such as segmentation, feature extraction, and classification, are employed to process images captured by video cameras [10]. It is more popular due to its advantages, including fast response times and lower costs [13].

In this article, a novel detection technique is proposed to identify two categories of traffic violators: those who violate the signal lane and those who are motorcyclists without helmets [15]; [17]. This methodology utilises various algorithms, such as Faster R-CNN, to track the vehicle object, recognise traffic rule violations, and detect the license plate of the violating vehicle. For vehicle object detection, the Region Proposal Network (RPN) and Fully Convolutional Neural Network (FCNN) work together as a combined process within Faster R-CNN [18]. In our application, Faster R-CNN is used to classify the types of traffic violations occurring [16]. Finally, number plate recognition is processed using the standard ALPR approach, which is supported by a character recognition technique [19]. After segmentation, the alphabet and numbers in the number plate image are identified through character recognition [24]. Three different activities are successfully recognised by a single automated system, making this method superior to other existing literature [22]. The remaining section of the paper is arranged as follows: The literature related to our proposed technique is presented in Section II. Section III provides a detailed explanation of the detection method and its stages. The results, following testing and discussions on the outputs, are presented in Section IV. Finally, section V enumerates the conclusion and plans [26].

2. Review of Literature

This section presents several automated traffic violation detection systems that employ different techniques. Shreyas et al. [21] proposed a system for automatic License Plate Recognition (ALPR) based on advanced image processing. This system is used to monitor road activities, including speeding vehicles and lane violations. This system detects the particular vehicle that violates the traffic rules, and from the captured image, the number plate area in the vehicle image is extracted alone. Later, the optical character recognition technique was used. Simulations are performed using the MATLAB tool; therefore, the vehicle number is provided to the relevant authority, and an SMS is sent to the user. Moreover, Celik and Kusetogullari [25] designed a real-time surveillance system to monitor vehicles that violate the maximum speed limit set by the authorities. A solar battery array supplies power to the detection system. Here, the sequence of motion is recorded by a video camera to construct the binary sequence [28]. Additionally, a novel adaptive thresholding method is introduced to digitise the outputs for estimating the speed of moving vehicles [20]. Furthermore, an analysis is conducted to identify the number plate image of a specific vehicle. In addition, Singh et al. [23] introduced a Radio Frequency Identification (RFID) system for detecting speeding vehicles.

This system automatically sends the identity to the remote device. Here, one or more transponders are used to store the data or transfer the data. This technique can detect the vehicles of speed violators even in adverse weather and lighting conditions. He et al. [9] and Meiring and Myburgh [3] discussed various intelligent driving style analysis systems and related Artificial Intelligence (AI) algorithms. In this survey, various driving styles are analysed to identify the current behaviour of any driver. To perform this function, various AI algorithms and machine learning approaches are applied. Additionally, various solutions based on each algorithm are analysed to determine the most straightforward model.

In a paper, Wu et al. [11] presented a technology for traffic sign recognition. According to this method, the features of the traffic sign image are extracted using a Convolution Neural Network (CNN). Then the feature map is constructed and subjected to filtration and regression by a Region Proposal Network. Later, the Regions are mapped to the feature map through a Region of Interest (ROI) layer. Finally, by using the classification network, the specific classification task is performed. Ki and Baik [29] enumerated a deep learning technique for recognising a variety of number plates in Indian vehicles. The system utilised a Faster-Regional CNN, which provides an effective solution for recognising number plates with various irregularities in India. This method has achieved an overall precision of 94.98% for all types of vehicles.

Donoser et al. [12] developed a sensorless detection system using several image processing techniques. The developed system will detect vehicles violating the red traffic light. If any vehicle fails to stop when the red light is on, it is detected through a single camera frame using various techniques. Once the non-stopped vehicle image is captured, the number plate is tracked across different video frames, and the corresponding picture frame is stored. The primary drawback of this methodology is the potential for false triggering of the camera, which can result in the detection of vehicles that are not actually present. A complete detection system in traffic control is introduced by Wang et al. [27]. Accordingly, an improved algorithm called background updating is applied using the wavelet transform. In this approach, a feature-based tracking method is used to track the moving vehicles. Thus, a comprehensive traffic violation detection system has been developed using C++ with OpenCV.

3. Proposed System

In this section, the detailed process for the traffic violation detection system is explained. This procedure is carried out in three stages: (1) Vehicle object tracking, (2) Violation identification, and (3) License plate recognition. Figure 1 illustrates the functional block diagram of our proposed system. The detailed procedure in each stage is explained further.

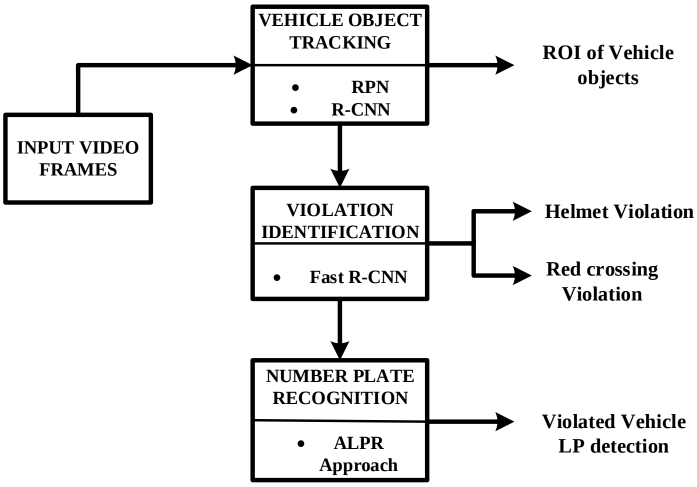

Figure 1: Functional block diagram of the proposed approach

3.1. Vehicle Object Tracking

Faster R-CNN is a deep learning-based technique. This Faster R-CNN is a combination of two modules, the Region Proposal Network (RPN) and R-CNN. The two modules perform two important roles: (1) RPN is used to extract the ROI, and F-CNN helps to generate the proposed ROI using a convolutional neural network. Initially, an RPN network is designed, and then an algorithm is developed to train both modules. The performance of the proposed method is also compared to that of alternative models, namely SSD and YOLO. Figure 2 shows the functional diagram of Faster R-CNN. The two popular modules, RPN and R-CNN, are discussed in the subsections that follow.

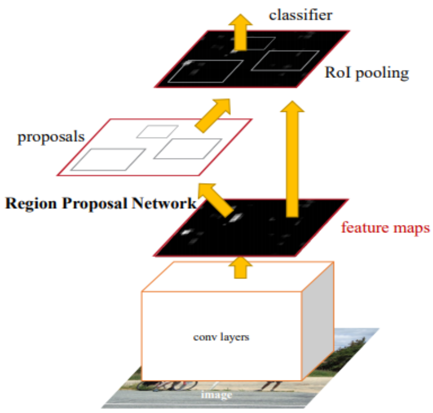

Figure 2: Functional diagram of Faster R-CNN

3.1.1. Region Proposal Network

RPN belongs to the Fully Convolutional Neural Network (FCNN) category [16], which is capable of predicting object bounds in an image and the object scores at each position. Specifically, during training of the RPN, it generates high-quality region proposals. A wide range of scales and aspect ratios is brought using the anchor boxes. In RPN, it takes the video frame image as input and produces a set of rectangular boxes as output proposals. A window size of 5 x 5spatial slides from the beginning of the feature map, which is taken from the last shared convolutional layer. This sliding window is mapped to a lower-dimensional feature map, which is then fed to a fully connected layer comprising a bounding-box regression layer (reg) and a bounding-box classification layer (cls). The working of RPN is represented in Figure 3.

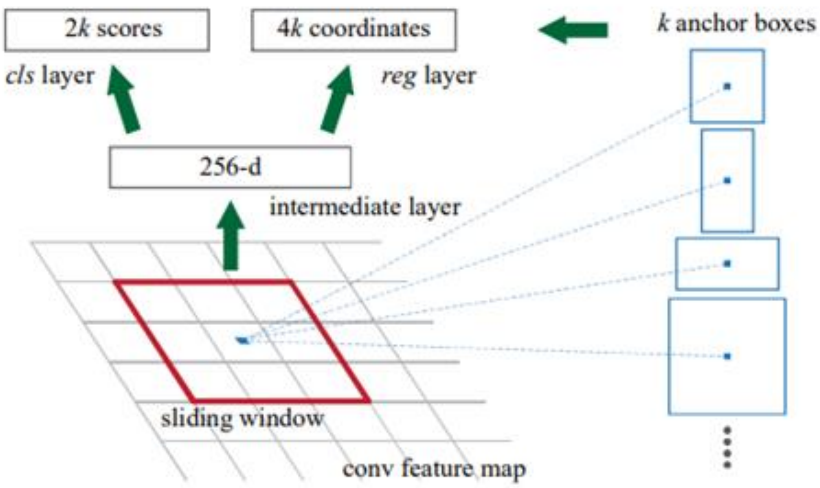

Figure 3: Working of an RPN

In our case, if k vehicle objects are predicted in the video frame at each sliding window location, it is shown in Figure 3. Normally, the predicted k proposals are referred to as anchors. For the following three aspect ratios (1:1, 1:2, and 2:1), each sliding window yields a total of k = 7 anchors. In the regression layer, there are 4k outputs, and in the classification layer, there are 2k scores. The total size of the convolutional feature map is represented by (W × H) × k anchors.

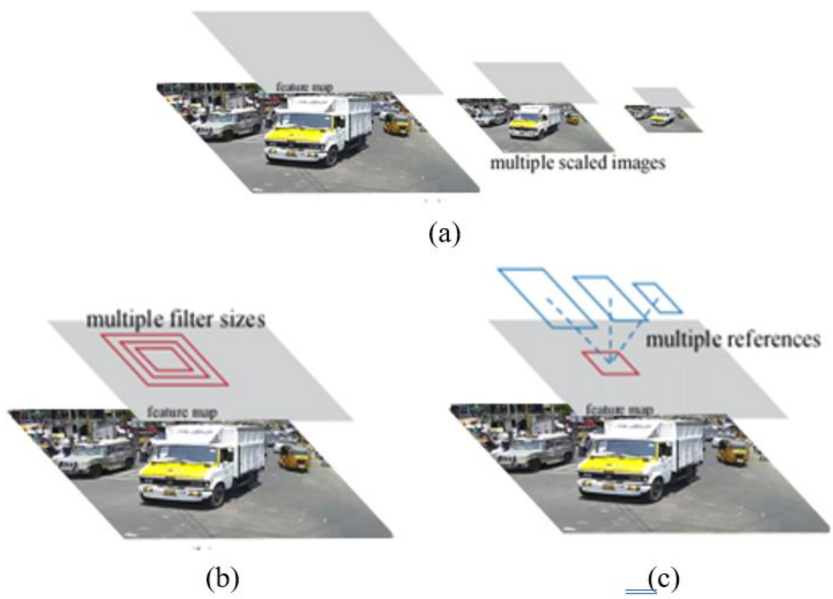

Figure 4: Multi-scale pyramid of anchors (a) Image of feature map (b) Filters with multiple scales (c) Reference boxes in regression function

The design of multi-scale anchors is crucial for addressing the various scales and sharing common features across them. Therefore, a new method is introduced to address the multiple scales, as depicted in Figure 4. Here, a multi-scale anchor is developed with a pyramid of anchors. As a result, a regression output is obtained in bounding boxes, referencing the anchor boxes of various aspect ratios and multiple scales. This output depends on the images and feature maps such that a single size filter is used. The scheme for addressing multiple sizes and scales is tabulated in Table 1.

Table 1: Different settings for anchor scales and aspect ratio

Settings	Anchor scales	Aspect Ratio
1 scale one ratio	128^2, 256^2	{1:1, 1:1}
1 scale three ratio	128^2, 256^2	{2:1, 1:1, 1:2} {2:1, 1:1, 1:2}
3 scale one ratio	{128^2, 256^2, 512^2}	{1:1}
3 scale three ratio	{128^2, 256^2, 512^2}	{2:1, 1:1, 1:2}

It is necessary to assign each anchor a labelled binary class to determine whether it represents a vehicle object or not. Usually, the anchor is assigned based on the highest overlap ratio (IoU). Suppose the overlap ratio is > 0.7 for all ground-truth boxes; it is labelled as positive. Likewise, if the overlap ratio is < 0.3 for all ground-truth boxes, it is labelled as negative. Moreover, a single ground truth box is assigned to multiple anchors and is labelled as positive. The loss function for an image is given by:

$$L(\{P_i\},\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}(p_i, p_i^*) + \boldsymbol{\lambda}\frac{\mathbf{1}}{\mathbf{N_{reg}}}\sum_{\mathbf{i}} p_i^* L_{\mathbf{reg}}(x_i, x_i^*) \tag{1}$$

Where,

i- denotes the mini-batch index of an anchor.
p_(i)-represents the predicted object probability of an anchor i.p_i^* - stands for ground-truth box (1 or 0) for positive/ negative label anchor.x_irepresents the parameterised vector coordinates of the predicted bounding-box.
x_i^*-indicatethe coordinates of the positive label ground-truth box.
L_cls-represents the classification loss over the function.
L_reg-represents the regression loss over the function.
p_i^* L_reg-indicates the positive anchors regression loss.
N_cls-Normalised classification loss.
N_reg-Normalised regression loss.
λ-is the balancing parameter used as a weighted function for regression loss.
For bounding-box regression, the four coordinates are calculated from the expressions (2)-(9).

$$r_x = {(a - a_i)}/{x_a} \tag{2}$$

$$r_y = {(b - b_i)}/{k_a} \tag{3}$$

$$r_w = \log\left(\frac{x}{x_a}\right) \tag{4}$$

$$r_k = \log\left(\frac{k}{k_a}\right) \tag{5}$$

$$r_x^* = {(a^* - a_i)}/{x_a} \tag{6}$$

$$r_y^* = {(b^* - b_i)}/{k_a} \tag{7}$$

$$r_w^* = \log\left(\frac{x^*}{x_a}\right) \tag{8}$$

$$r_k^* = \log\left(\frac{k^*}{k_a}\right) \tag{9}$$

Where (x, y) indicate the centre coordinates of the anchor, w indicates the width, and k indicates the height. In addition, a ' stands for predicted box, ai denotes the variable for anchor box, and a∗ represents the variable of ground truth box, respectively. Later, the RPN network is trained using back propagation and Stochastic Gradient Descent (SGD) algorithms [24]. Again, the network is trained using an image-centric sampling strategy, which is proposed in Ren et al. [22]. Based on the sampling method, 256 anchors in an image are randomly sampled to optimise the loss function. Here, we maintain a 1:1 ratio for both positive and negative anchors. If the image consists of 128 or fewer samples, the mini-batch is padded with negative anchors.

Figure 5: Sample output of vehicle object tracking

To obtain a zero-mean Gaussian distribution, the drawing weights are randomly tuned from 0.01 to 1. A pre-training model initialises all the layers in ImageNet classification [25]. To optimise the memory, the layers of the ResNet are tuned on the traffic dataset. The learning rate parameters are chosen as follows: for 2k mini-batches, 0.001, and for 0.5k mini-batches, 0.0001. Another required field, such as momentum rate, is fixed at 0.85, and the weight decay is 0.0006. Once the RPN is trained for generation, Faster R-CNN is implemented for the detection using the proposals. In the proposed algorithm, a technique is designed to learn the shared features. The CNN is formed by combining the RPN and Faster R-CNN as a unified learning network. Figure 5 shows the sample output of vehicle object tracking.

3.1.2. Training Algorithm:

Step 1: Collect the data and preprocess the data
Step 2: Design the network architecture by adding the RPN layers on the base CNN networks.
Step 3: Generate a set of anchors at each sliding window position on the feature map.
Step 4: Train the RPN by assigning a binary label, defining the loss function, and optimising it.
Step 5: Handling the imbalance present in the positive and negative samples within each mini-batch.
Step 6: Initialise the pretrained weights, tune the learning rate parameters, and evaluate the model.
Step 7: Apply non-maximum suppression to remove the redundant losses.
Step 8: The generated RPN proposals are fed to Faster R-CNN to classify the bounding boxes.

3.2. Violation Identification

To analyse the red signal crossing violation, the marking lines, such as stop lines, edge lines, and double lines on a straight road, should be verified (Figure 6).

Figure 6: Training image sample in the violation class

In this work, the image frame is pre-processed by an edge detector algorithm, and then, by applying the Hough transform [29], the background lines are identified (Figure 7).

Figure 7: Training image sample in non-violent class

After the detection of lines, the traffic light colour is selected to indicate whether a crossing of that line occurs at a particular moment (Figure 8).

Figure 8: Helmet detection

Thus, the red signal crossing violation is identified using Faster R-CNN (Figure 9).

Figure 9: Helmet violation identification

The line detection, red signal non-violation, and violation results are presented in Figure 10.

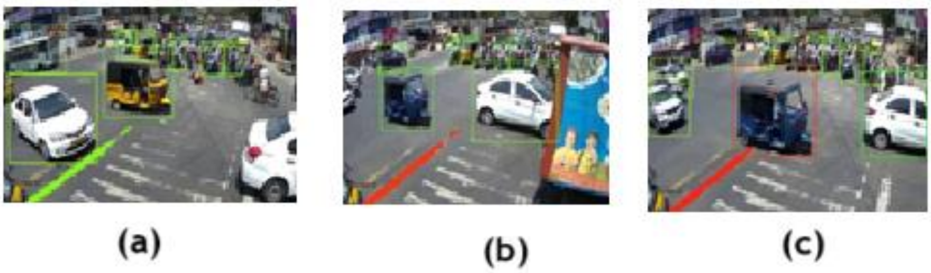

Figure 10: Red signal crossing violation (a) line detection (b) non-violation of red signal crossing (c) violation of red signal crossing

3.3. License Plate Recognition

In this phase, the number plate recognition technique and the stages involved in the ALPR are briefly discussed. The process of ALPR is divided into five subsections: (1) LP detection, (2) character segmentation, (3) character recognition, (4) majority voting, and (5) LP recognition. Figure 11 illustrates the proposed ALPR approach. Different CNN models are adopted in each stage of the ALPR approach. The Faster R-CNN model is applied for LP detection and character segmentation, and a CNN is used for character recognition. In the above stages, the tuning operation is performed at each stage to ensure the task is completed without error.

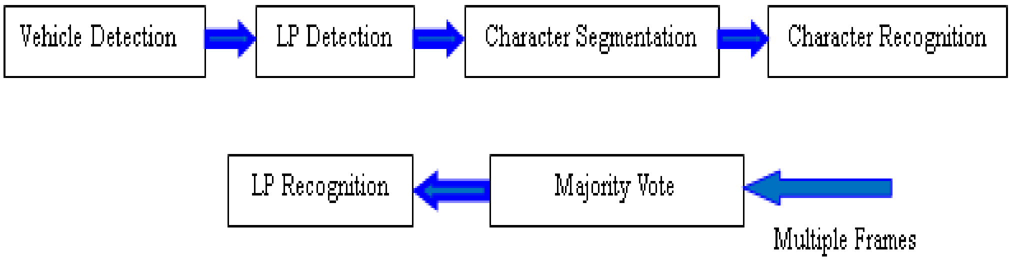

Figure 11: Pipeline of proposed ALPR approach

- **Stage 1:** Firstly, the License plate detection must be performed. In different countries, license plates come in various models, shapes, and sizes, most commonly rectangular or trapezoidal. In a few vehicles, the license number is painted or engraved in the space provided. Similarly, the mean size of the plate also varies from vehicle to vehicle, from 200 mm x 100 mm up to 500 mm x 120 mm. By considering all the issues, the best approach is to label a rectangular bounding box for each plate. In our proposed scheme, FastR-CNN is used to train the network separately for both the VGG16 and ResNet-50 models.
- **Stage 2:** In the second stage of the ALPR approach, character segmentation is performed, which is crucial for vehicle recognition. Actually, the characters present in the number are non-identical in terms of font, size, and padding. Therefore, the open type of Lecun et al. [28] cannot be implemented. In such a case, the plates should be labelled to train the network and identify the location of characters.
- **Stage 3:** In the third stage, character recognition is performed to recognise the characters, such as letters or numbers, present in the cropped images. The standard CNN is applied. The letters are cropped from the plates, and particular letters are taken for training the characters. Around 1000 characters are taken from different plates for training the model. An additional 500 characters are utilised for further training. A total of 1,500 characters, with 100-200 characters per class, are subject to training. To achieve better results, recursive training is executed. Anyway, some identical characters, such as S and 5, O and Q, one and I, etc., with similar features, may find complications during detection.
- **Stage 4:** The confusion problems can be solved only through the majority voting technique. In this technique, the final recognition primarily depends on the character that is frequently predicted at each LP position. Temporary information is previously stored in Wang et al. [27] and Lecun et al. [28]. As a result, the majority voting stage has greatly increased the recognition rates. The results of LP detection and character segmentation for various vehicle types are presented in Figure 12.

Figure 12: Results of LP Detection and Character segmentation for different types of vehicles

4. Experimental Results

Based on the workflow, the test results of the proposed system are obtained. The results are extracted at different stages of traffic violation detection. The output images at different stages, like vehicle object tracking, violation classification, and number plate detection, are obtained. The estimated measures at each level are presented in the subsections below:

4.1. Vehicle Detection and Classification Results

Initially, the input video frame of size 1280×720 with RGB colour input channel is fed to the vehicle object tracking and classification system. To perform this function, RPN and Faster R-CNN techniques are applied. The Indian traffic dataset is utilised to validate the vehicle object tracking system. In the Indian dataset, the vehicle annotations are not provided. Therefore, the annotations are manually labelled within the vehicle bounding box of each image in the dataset. The results are evaluated, and metrics such as sensitivity, specificity, and Accuracy Are Considered (Figure 13). These expressions used to evaluate the metrics are given in equations (10)-(12).

$$\text{Sensitivity} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \tag{10}$$

$$\text{Specificity} = \frac{\text{FN}}{(\text{FN} + \text{FP})} \tag{11}$$

$$\text{Accuracy} = \frac{(\text{Sensitivity} + \text{Specificity})}{2} \tag{12}$$

Table 2: Comparative results of vehicle object detection

Methods	Sensitivity	Specificity	Accuracy	Runtime(sec)
SSMD	91.8%	91.1%	91.3%	63
YOLOv2	94.1%	93.6%	93.7%	49
RCNN	94.6%	94.2%	94.5%	55
F-RCNN	96.2%	95.6%	95.8%	38

Number of data=1000

An analysis is conducted to compare the proposed system with previous techniques [20]; [21]; [19].

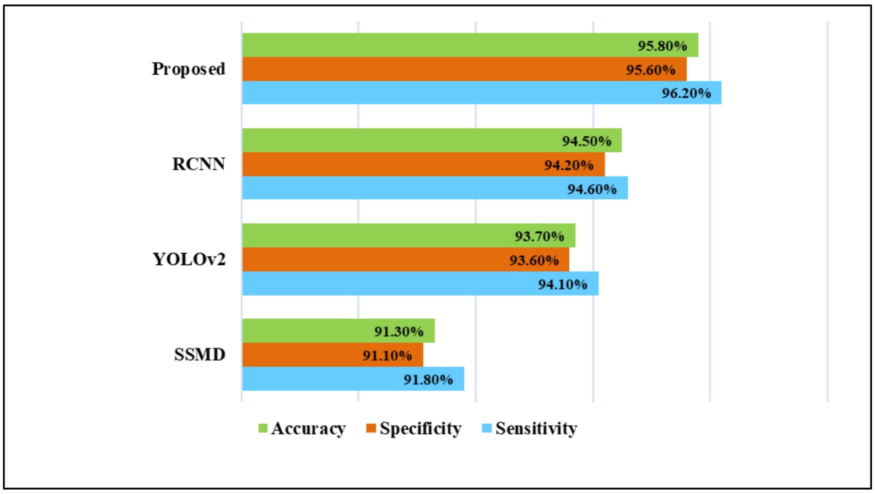

Figure 13: Comparative result analysis in vehicle object detection

Table 2 presents the results obtained and previously published using different methods. Accordingly, our proposed method proved to be robust for detection and recognition using the Indian traffic dataset (Figure 14).

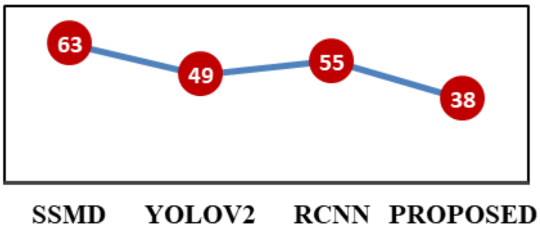

Figure 14: Comparative analysis of runtime vehicle object detection (in seconds)

4.2. Traffic violation results

Under traffic violation detection, two cases, such as Helmet Detection and Red Signal Crossing Detection, are performed separately. The proposed model is trained using less data, which reduces the computational burden of the algorithm. Numerous experiments were conducted to determine the most effective method for detecting traffic violations and minimising false detections. Although we found a few misdetections, the error rate has been dramatically reduced. The comparative results of helmet violation detection are tabulated in Table 3, and the results of red signal crossing violation are presented in Table 4. The evaluation matrices, like precision, recall and F1 score, are determined using the following expressions.

$$\text{Precision} = \frac{\text{TP}}{(\text{TP+FP})} \tag{13}$$

$$\text{Recall} = \frac{\text{TP}}{(\text{TP+FN})} \tag{14}$$

$$\text{F1 score} = \frac{2\times\text{precision}\times\text{recall}}{(\text{precision+recall}} \tag{15}$$

Table 3: Comparative results of helmet detection

Method	Precision	Recall	F1 Score
HOG	0.96	0.72	0.84
HAAR	0.97	0.78	0.88
Faster-RCNN	0.98	0.89	0.94

Where TP denotes true positive boundary box LPs detected correctly, FP denotes the false positive boundary box LPs, and FN represents false negative LPs detected incorrectly.

Table 4: Comparative results of red signal crossing violation

Method	Precision	Recall	F1 Score
YOLOv2	0.98	0.89	0.93
Faster-RCNN	0.99	0.95	0.97

Here, the proposed Faster R-CNN method is analysed in comparison to other versions based on Girshick [19] and Sermanet et al. [18]. It is found that the helmet detection system using HOG is not robust under a changing environment. If the location of training and testing differs or if the object changes shape, the accuracy of detection is not very convincing. Moreover, if the scenic view changes, the cascade of HOG and Haar cascades are more sensitive. However, in the case of the Faster R-CNN model, this does not occur. Furthermore, the response of the Faster R-CNN model can be further improved by increasing the number of training sets.

4.3. ALPR Results

The evaluation of LP detection and character recognition is done by calculating Precision and Recall. Essentially, ResNet50 and VGG16 are selected as the Base CNN for Faster R-CNN, specifically for LP detection and character recognition. The experimental results of precision and recall are given in Table 5.

Table 5: Results of LP and CS for base CNN with RESNET50 and VGG16

Network	Base CNN	Precision	Recall
Number Plate Detection (NPD)	ResNet-50	95.38%	92.99%
	VGG16	91.97%	89.97%
Character Segmentation (CS)	ResNet-50	97.42%	95.62%
	VGG16	93.25%	91.55%

The accuracy of character recognition is calculated by the ratio of the number of characters recognised correctly to the total number of ground truth characters. Table V reveals that, when ResNet50 is used as the base CNN for Faster R-CNN, its performance is significantly better than that of VGG16.

5. Conclusion

In this paper, we present a deep learning-based automated system for detecting traffic violations involving vehicles. For this purpose, the Faster R-CNN is mainly used. The traffic-violating vehicles are detected using the proposed techniques from the video input frames captured from a fixed-position video camera. The helmet-violating vehicles and red signal-running vehicles are detected and tracked, and their license plate regions are subsequently traced, with the information being transferred to the relevant authorities for further action. The proposed system is tested with the Indian traffic signals public data set, which shows an accuracy of 95.8% in vehicle object detection. Moreover, the system has reached 98% precision in helmet violation and 99% in red signal crossing violation. Furthermore, the proposed method achieved 95.38% and 91.97% accuracy using ResNet and VGG16 as the Base CNN in license plate detection.

Acknowledgement: The authors collectively acknowledge the support and cooperation of their Universities and Colleges in facilitating this research. All contributors have equally participated and approved the final version of the manuscript.

Data Availability Statement: The study utilises a dataset focused on real-time detection of traffic violations using deep learning methodologies. All authors confirm that the data supporting the findings of this research are available upon reasonable request.

Funding Statement: The authors jointly declare that no external or institutional funding was received in the preparation of this research or manuscript.

Conflicts of Interest Statement: All authors affirm that there are no conflicts of interest related to this study. References and sources have been properly cited throughout the work.

Ethics and Consent Statement: Ethical approval was granted by the respective institutions, and informed consent was obtained from all participants and organisations involved. Each author contributed significantly and approved the final version of this manuscript.

References

1. C. B. Crawford and V. P. A. Doran, "Trial and evaluation of intelligent road studs for hazard warning," *in Proc. 12th IEE Int. Conf. Road Transp. Inf. Control (RTIC 2004)*, London, United Kingdom, 2004.
2. C. Szegedy, S. Reed, D. Erhan, and D. Anguelov, "Scalable, high-quality object detection," *arXiv preprint arXiv:1412.1441v2*, 2015. Available: https://arxiv.org/abs/1412.1441v2. [Accessed by 12/09/2022].
3. G. A. M. Meiring and H. C. Myburgh, "A review of intelligent driving style analysis systems and related artificial intelligence algorithms," *Sensors (Basel)*, vol. 15, no. 12, pp. 30653–30682, 2015.
4. G. R. Gonçalves, D. Menotti, and W. R. Schwartz, "License plate recognition based on temporal redundancy," *in Proc. IEEE 19th Int. Conf. Intelligent Transportation Systems (ITSC)*, Rio de Janeiro, Brazil, 2016.
5. H. Cheng, B. Shaw, J. Palen, B. Lin, B. Chen, and Z. Wang, "Development and field test of a laser-based nonintrusive detection system for identification of vehicles on the highway," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 2, pp. 147–155, 2005.
6. J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," *in Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Montreal, Quebec, Canada, 2015.

7. J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” *in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, Massachusetts, United States of America, 2015.
8. K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, Nevada, United States of America, 2016.
9. K. He, X. Zhang, S. Ren, and J. Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” *in Proc. European Conf. Computer Vision (ECCV)*, Zurich, Switzerland, 2014.
10. L. Lin, X. B. Han, R. Ding, G. Li, C.-Y. Lu, and Q. Hong, “A new rechargeable intelligent vehicle detection sensor,” *J. Phys., Conf. Ser.*, vol. 13, no. 9, pp. 102–106, 2005.
11. L. Wu, H. Li, J. He, and X. Chen, “Traffic sign detection method based on Faster R-CNN,” *J. Phys. Conf. Ser.*, vol. 1176, no. 3, pp. 1-6, 2019.
12. M. Donoser, C. Arth, and H. Bischof, “Detecting, tracking and recognizing license plates,” *in Computer Vision – ACCV 2007*, *Springer*, Berlin, Germany, 2007.
13. M. Garrido and M. Sotelo, “Fast road sign detection using Hough transform for assisted driving of road vehicles,” *in Lecture Notes in Computer Science, Springer*, Cham, Switzerland, 2014.
14. N. S. Bayindir, H. Demirel, and H. Kusetogullari, “A sensorless red light violation detection system using image processing,” *Journal of Electronics World and Wireless World*, vol. 113, no. 1859, pp. 32–34, 2007.
15. O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, “ImageNet large scale visual recognition challenge,” *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.
16. P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, “Object detection with discriminatively trained part-based models,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.
17. P. Ravirathinam and A. Patawari, “Automatic license plate recognition for Indian roads using faster-RCNN,” *in 2019 11th International Conference on Advanced Computing (ICoAC)*, Chennai, Tamil Nadu, India, 2019.
18. P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, “OverFeat: Integrated recognition, localization and detection using convolutional networks,” *arXiv preprint arXiv:1312.6229*, 2013. Available: https://arxiv.org/abs/1312.6229. [Accessed by 08/03/2023].
19. R. Girshick, “Fast R-CNN,” *in Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, 2015.
20. R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” *in 2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, Ohio, United States of America, 2014.
21. R. Shreyas, P. V. Pradeepkumar, H. B. Adithya, B. Padmaja, and M. P. Sunil, “Dynamic traffic rule violation monitoring system using automatic number plate recognition with SMS feedback,” *in Proc. 2nd Int. Conf. Telecommun. Netw., NITTTR*, Chandigarh, India, 2018.
22. S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” *in Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Montreal, Quebec, Canada, 2015.
23. S. Singh, R. Chawla, and H. Singh, “Intelligent speed violation detection system,” *Int. J. Eng. Techn. Res. (IJETR)*, vol. 2, no. 1, pp. 92–95, 2014.
24. S. Zander, T. Nguyen, and G. Armitage, “Automated Traffic Classification and Application Identification using Machine Learning,” *in Proceedings of the IEEE Conference on Local Computer Networks 30th Anniversary*, Sydney, New South Wales, Australia, 2005.
25. T. Celik and H. Kusetogullari, “Solar-powered automated road surveillance system for speed violation detection,” *IEEE Trans. Ind. Electron.*, vol. 57, no. 9, pp. 3216–3227, 2010.
26. World Health Organization, “Global Status Report on Road Safety,” *World Health Organization*, 2013. Available: http://www.who.int/violence_injury_prevention/road_safety_status/2013/en/. [Accessed by 02/06/2021].
27. X. Wang, L.-M. Meng, B. Zhang, J. Lu, and K.-L. Du, “A video-based traffic violation detection system,” *in Proceedings 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC)*, Shenyang, China, 2013.
28. Y. Lecun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, “Backpropagation applied to handwritten zip code recognition,” *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.
29. Y.-K. Ki and D.-K. Baik, “Model for accurate speed measurement usingdouble-loop detectors,” *IEEE Transactions on Vehicle Technology*, vol. 55, no. 4, pp. 1094–1101, 2006.