Extra Papers for Literature Review

Contents

[**Automatic Traffic Rule Violation Detection and Number Plate Recognition** 2](#_Toc29805313)

[**Algorithm for detecting violations of traffic rules based on computer vision approaches** 2](#_Toc29805314)

[**Traffic Violation Detector using Object Detection** 3](#_Toc29805315)

[**Vehicle Detection and Tracking from Video Frame Sequence** 3](#_Toc29805316)

[**Over speed detection using Artificial Intelligence** 3](#_Toc29805317)

[**Automated High-Speed Traffic Monitoring and Violation Detection Using RFID Technology** 5](#_Toc29805318)

[**V-ITS: Video-based Intelligent Transportation System for Monitoring Vehicle Illegal Activities** 5](#_Toc29805319)

[**Police Eyes: Real World Automated Detection of Traffic Violations** 6](#_Toc29805320)

[**Real-Time Vehicular Traffic Violation Detection in Traffic Monitoring Stream** 7](#_Toc29805321)

[References 8](#_Toc29805322)

# **Automatic Traffic Rule Violation Detection and Number Plate Recognition**

The project presents Automatic Number Plate Recognition (ANPR) techniques and other image manipulation techniques for plate localization and character recognition which makes it faster and easier to identify the number plates. After recognizing the vehicle number from number plate the SMS based module is used to notify the vehicle owners about their traffic rule violation. An additional SMS is sent to Regional Transport Office (RTO) for tracking the report status. Three algorithms that are used are: Edge Detection, Hough Transform & K-Nearest Neighbor. For detection of the number plate, edge detection is first of all performed, a Gaussian smoothed step edge is used for the detection. This produces a noisy edged picture of the subject. After that for clearing of noise, Hough Transform is used. Hough Transform also helps in line detection. The k-nearest neighbors formula (k-NN) may be a non-parametric methodology used for classification and regression. It is implemented and executed in OpenCV and performance is

tested on genuine images. The system works quite well. [1]

**Algorithm for detecting violations of traffic rules based on computer vision approaches**

The algorithm uses multi-step proceedings. For car detection, we use faster R-CNN deep

learning tool. The algorithm shows promising results in the detection violations of traffic rules. The algorithm works in two stages. At the first stage, it is necessary to detect three classes of objects in the video sequence. First class crosswalk (zebra), the second class moving vehicles and the third class the pedestrians going on the crosswalk. After all the object classes are found on each frame of the video sequence, the second stage begins. It is necessary to impose video sequence frames with the allocated objects at each other. If on the same frame of the video sequence the pedestrian and the vehicle are at the crosswalk, the vehicle means violates Traffic regulations, having done not pass the pedestrian. The background is received using a segmentation method. It is the previously discussed background difference method. Car detection is performed using Faster R-CNN Deep Learning. Create a Convolutional Neural Network (CNN). A CNN is the basis of the Faster R-CNN object detector. Beginning with the image input layer function, which defines the type and size of the input layer. For classification tasks, the input size is typically, the size of the training images. For detection tasks, the CNN needs to analyze smaller sections of the image, so the input size must be similar in size to the smallest object in the data set. In this data set all the objects are larger than [16 16], so select an input size of [32 32]. This input size is a balance between processing time and the amount of spatial detail the CNN needs to resolve. The middle layers are made up of repeated blocks of convolutional, ReLU (rectified linear units), and pooling layers. These layers form the core building blocks of convolutional neural networks. Combine the input, middle, and final layers. The system trains the detector in four steps. The first two steps train the region proposal and detection networks used in Faster R-CNN. The final two steps combine the networks from the first two steps such that a single network is created for detection [2]. Each training step can have different convergence rates, so it is beneficial to specify independent training options for each step. Now that the CNN and training options are defined train the detector using training Faster RCNN Object Detector. During training, image patches are extracted from the training data. The 'PositiveOverlapRange' and 'NegativeOverlapRange' name-value pairs control which image patches are used for training. Pedestrians are detected using Motion-Based Multiple Object Tracking of a method. Detection of moving objects uses the algorithm of

subtraction of a background based on a Gaussian mixture of the model. The movement of each

traced object is estimated using Kalman's filter.[2]

# **Traffic Violation Detector using Object Detection**

Traffic Violation Detector using Object Detection helps to detect the vehicle number plate that is violating traffic rules and by that number the admin finds the details of the car owner and penalty is sent to the violator. This system realizes tensorflow, which makes it easy to construct, train, and deploy object detection models. Generative Adversarial Network (GAN), which is a deep learning algorithm, is used to model or generate data that is very similar to the training data. It consists of a discriminator to identify if the data is real or fake. GAN is based on Deep Learning Techniques. COCO (Common Object in Context) is dataset containing 200,000 images and more than 500,000 object annotations in 80 different categories. This dataset is used in the system. The process sequence is: Image Classification, Image Tagging, Object Detection, Optical Character Recognition and Image Segmentation. The Nanonet API is used for Optical Character Recognition (OCR). [3]

# **Vehicle Detection and Tracking from Video Frame Sequence**

In the pre-processing phase of this paper, the noises are denoised with the help of filtering techniques. After it, the segmentation is done using regionprops function of MATLAB. The =n background detection is done, after which feature extraction is performed, which extracts features such as image edges, corners, and other structures. After that, the background difference algorithm is applied. The vehicle gets detection. Now, for motion estimation, Optical flow method is applied. The Optical Flow block using the Horn – Schunck algorithm (1981) estimates the direction and speed of object motion from one video frame to another and returns a matrix of velocity components. Various image processing techniques such as thresholding, median filtering are then sequentially applied to obtain labeled regions for statistical analysis. Vehicles are also tracked using the method. [4]

# **Over speed detection using Artificial Intelligence**

In this project, by the use of computer vision and artificial intelligence, over speeding is tried to be detected and reported to the violation to the law enforcement officer. It was observed that when predictions are done using YoloV3, the best results were obtained. Artificial intelligence can be defined as a science or engineering of making machines smart and intelligent. Deep Learning is a part of artificial intelligence which primarily deals with the neural networks. Neural networks try to learn from the training data without being programmed explicitly. Different functions are used for calculation of losses. The back-propagation algorithm is carried so as to update the weights. A traffic signs classifier is implemented using pandas, NumPy, SkImage, SkLearn, h5py, glob, Keras, Matplot lib, OpenCV and python. It took a little more than 9 hours to train the model. Over the whole course of training, it was observed that loss was decreasing with increase in the epochs. Initially, loss was very high and was decreasing non-linearly. After 30 epochs there was a negligible change in the loss and therefore training the model for 30 epochs would be the most efficient. The best accuracy the system was able to achieve was 96% on the validation set. the systemt takes ~2 seconds for inference of single image using the trained model on the same machine. For testing, the system had clicked manually 25 images and these images were cropped so just as to get the traffic sign. When testing was done over these images, 21 images predicted the correct categories whereas 4 images predicted the invalid categories. Tools used to achieve image data augmentation consists of MATLAB, python, SkiImage, NumPy and OpenCV (Computer Vision library). These algorithms work for a constrained environment and fail if the images have variations. They may be able to detect a single large-sized soccer ball in the image very accurately but won’t work if we wish to detect many small size soccer balls of different variations present in the image. Under this kind of situations, Yolo comes to rescue. Yolo is an object detection system and is able to detect a wide variety of the objects present in the real time. Because of its unified architecture, it is extremely fast in detection. Existing deep learning classifier models like Regions with Convolutional Neural Network (R-CNN) are capable of performing object detection. For object detection, these systems use sliding window i.e they consider a classifier for every object to be detected and slide it over all possible window locations on the image. Once the classification is done, post-processing is carried out and bounding boxes are redefined. Post processing is also done to remove duplicated detections. This increases the complexity, computation and time it takes for the detection. The basic motivation for using Yolo is the speed and complexity of the system. Instead of sliding over an image many times, Yolo only looks once and detects all the objects present in the image. Yolo defines the detection problem as a regression problem and uses features from an entire image at the time of training. Unlike RCNN, it looks at the entire image during the time of 27 training and testing That means Yolo predicts all the different object categories present on an image simultaneously. Yolo uses non-maximal suppression [N13]. Neural network of Yolo is very similar to. GoogLeNet has 22 convolutional layers whereas Yolo has 24 convolutional layers. Convolution layer in Yolo is followed by the two fully connected layers. Size of the kernel used in convolution layers is 3\*3 or 5\*5. This causes the weights of convolution layer to be less dependent on the location of the objects in the image and weights do not have spatial information. Fully connected layer takes into consideration spatial far-away features. The system was finally able to achieve accuracy of around 90% for the images in day time but accuracy reduces if it is night time. [5]

# **Automated High-Speed Traffic Monitoring and Violation Detection Using RFID Technology**

The vehicle Identification is mainly done with special RFID readers. The main challenge is to be able to read RFID tags of high-speed vehicles. RFID readers are used for high-speed trains. Traffic monitoring is a wide area in research. Several methods exist for efficient real-time monitoring of the traffic, including RFID-based and CCTV camera-based systems. Several architectures for RFID-based systems have been proposed and utilized for traffic monitoring in some countries. As far as we know, none of the existing implementations utilize the RFID technology for traffic violation detection in high-speed roads. In this paper, a new architecture of using RFID technology in high-speed roads (highways and freeways) is presented; its problems and challenges are analyzed and a number of solutions are proposed. The proposed architecture consists of four hierarchical levels: Vehicle Identification, Row-level Processing, Road-level Processing, and Control Center. Each subsystem processes data of the detected vehicle at a higher level and passes the processed data to the next subsystem in a hierarchical manner. Also in this paper three sample algorithms are proposed in order to show how the architecture works in detecting traffic violations. The algorithm for speed limit violation detection uses the timestamp of two different positions, then calculates as well as compares the speed. Most of the traffic laws prohibit reverse move of the vehicles in highways and other roads. Detecting this violation is also very easy: if the two consecutive positions are in backward order, then the vehicle moved against the road direction. Here are many cases when the drivers try to enter the road from road exits (driving against the exit direction). This causes many accidents and should be detected. Adding a Vehicle Identification row in exits (or U-turns) makes the detection possible. If the vehicle enters the road from any exit, one of its positions will be in the exit and the next in the road. [6]

**V-ITS: Video-based Intelligent Transportation System for Monitoring Vehicle Illegal Activities**

Improved V-ITS system is developed in this paper to detect and track vehicles and driver’s activities during highway driving. This improved V-ITS system is capable to do automatic traffic management that saves traffic accidents. It provides the feature of a real-time detection algorithm for driver immediate line overrun, speed limit overrun and yellow-line driving. To develop this V-ITS system, a pre-trained convolutional neural network (CNN) model with 4-layer architecture was developed and then deep-belief network (DBN) model was utilized to recognize illegal activities. To implement V-ITS system, OpenCV and python tools are mainly utilized. The GRAM-RTM online free data sets were used to test the performance of V-ITS system. The overall significance of this intelligent V-ITS system is comparable to other state-of-the-art systems. The real-time experimental results indicate that the V-ITS system can be used to reduce the number of accidents and ensure the safety of passengers as well as pedestrians. We also discuss vehicle recognition and classification utilizing vehicle attributes like color, license plate, logo and type, provide a detailed description of the advances in the field. Deep Belief Network (DBN) classifier was added to pre-train CNN model for recognizing of driver’s illegal activities without using complex methods of image processing algorithms. The proposed V-ITS system outperforms compare to simple pre-train CNN model using PCA in the same selected dataset. [7]

# **Police Eyes: Real World Automated Detection of Traffic Violations**

The system relies crucially on motion analysis. Background subtraction is commonly used to extract moving object regions in image sequences. The problem they have encountered with image segmentation using background subtraction is that shadows of moving objects are included as part of the objects. Images are acquired from two IP cameras continuously. The low-resolution image from one camera is used for motion processing to detect blobs and to identify the point of a violation, whereas the high resolution image from the second camera is used to grab a snapshot of the violating vehicle. The images acquired from both cameras and a short video clip from the low resolution camera are saved to disk for evidence if a violation is detected. The video clip can be further used to aid in verifying the detected violation for non-trivial cases such as emergency lane changes, occlusion of the violating vehicle, and accidents. A Gaussian mixture model is used for every pixel in the image. The foreground image extracted from the background subtraction module is further processed for shadow pixels using our shadow detection algorithm. A combination of normalized cross correlation between the foreground region and the corresponding background pixels, along with RGB vector distances between the foreground pixels and underlying background pixels is used to identify shadow pixels. This approach complements the shadow detection method included in the background subtraction module and enables identifying shadows in some particular settings with weak shadows. Detection of shadow pixels depends on various parameters that must be tuned to work in different lighting conditions. The shadow detection module removes shadow regions considerably but also results in splitting of the actual blob when vehicle parts that are similar to the background are considered as shadow. Given a foreground image, foreground blobs are extracted through connected component analysis after performing morphological operations on the foreground image to remove noisy blobs. After the foreground blobs have been identified, the base profile is extracted for each blob. The base profile of a blob is the set of lowermost pixels of the external contour of the blob. After this, blob extraction is performed. Analysis of the region of intersection of the base profile of each blob with the violation area is used to detect violations. If a violation is detected, the system saves the current video frame from both the cameras along with the image identifying the point of violation. The point of violation is saved to disk along with the original image grabbed from the camera. [8]

# **Real-Time Vehicular Traffic Violation Detection in Traffic Monitoring Stream**

In order to achieve real-time analysis, parallel computing techniques are used in their implementation. An optimization scheme as well as a well design data structure is proposed to improve the performance of the parallel implementation. The system works on three violation aspects: speeding, black-listed and cloned plate vehicles. The system proposes an algorithm that can not only detect speeding and license plate cloning on roadways, but also in parking lots. An efficient data structure is proposed in the parallel implementation, which contributes to the high efficiency of the system. A multi-threaded optimizing scheme is proposed, which successfully reduces the synchronization overhead and achieve load-balance between multiple threads and processors. The system is implemented on the basis of real as well as synthetic data. [9]

# References

|  |  |
| --- | --- |
| [1] | A. Narkhede, V. Nikam, A. Soni, A. Sathe and G. G. Chiddarwar, "Automatic Traffic Rule Violation Detection and Number Plate Recognition," *Automatic Traffic Rule Violation Detection and,* pp. 559-563, Mar 2017. |
| [2] | S. Ibadov, R. Ibadov, B. Kalmukov and V. Krutov, "Algorithm for detecting violations of traffic rules based on computer vision approaches," in *MATEC Web of Conferences*, Rostov-on-Don, Russia, 2017. |
| [3] | S. Kasuadhan, "Traffic Violation Detector using Object Detection," SRMS College of Engineering,Technology & Research, Ghanghoua Ghanghori, India, Nov 2019. |
| [4] | M. Chaple and S. S. Paygude, "Vehicle Detection and Tracking from Video Frame Sequence," *International Journal of Scientific & Engineering Research,* vol. 4, no. 3, pp. 1-7, Mar 2013. |
| [5] | S. Patira, "Over speed detection using Artificial Intelligence," San Jose State University, San Jose, California, May 2019. |
| [6] | M. Mousaei, A. Keipour and E. Babaians, "Automated High-Speed Traffic Monitoring and Violation Detection Using RFID Technology," in *23rd Iranian Conference on Electrical Engineering*, Tehran, Iran, May 2015. |
| [7] | Q. Abbas, "V-ITS: Video-based Intelligent Transportation System for Monitoring Vehicle Illegal Activities," *International Journal of Advanced Computer Science and Applications (IJACSA),* vol. 10, no. 2, pp. 202-208, 2019. |
| [8] | R. Marikhu, J. Moonrita, M. Ekpanyapong, M. Dailey and S. Siddhichai, "Police Eyes: Real World Automated Detection of Traffic Violations," IEEE, Pathumthani, Thailand, 2013. |
| [9] | G. Ou, Y. Gao and Y. Liu, "Real-Time Vehicular Traffic Violation Detection in Traffic Monitoring Stream," in *2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, Beijing, China, 2012. |