SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM
TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE
Volume 8 No 5
ISSN 2054-7390

Monitoring Traffic Behaviors: Lane Detection and Speed Calculation

Saim Rasheed
Department of Information Technology, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Kingdom of Saudi Arabia;
srahmed@kau.edu.sa

ABSTRACT

In this work, we propose an image processing-based hybrid technique that provides assistance in detecting certain patterns in traffic infractions committed by drivers on the roads. The proposed technique is based on the speed estimation using video data in conjunction with tracking methods. Our hybrid proposal comprises two parts. First, we propose a method to detect the road lanes using Hough transform. Second, we detect the vehicles in the video datasets using Haar Cascade methodology and then track those vehicles for their speed estimation and for monitoring the driving patterns. In addition, the types of infractions that a driver can commit while driving are also detailed. The most significant cases in which the infractions are determined is when the driver makes a rapid and continuous change of lanes, parking at inappropriate places and prohibited U-turns. The results are provided in terms of vehicle detection and speed estimation. Our results and analysis reveal that the proposed method can assist in monitoring the driving patterns and detecting traffic infractions.

Keywords: Traffic behaviors; Driving patterns; Lane detection; Speed Calculation.

1 Introduction

In this era of modern technology, transportation vehicles, may they be public or private, play an important role in the development of the economy of any nation. The technology-based methods to improve road safety and ensuring uninterrupted flow of traffic has been a focus of research communities [1-5]. In everyday life, we can observe that the transport vehicles, either public transport or private vehicles, involve the participation of drivers and passengers. Regardless of the role that people are playing in a vehicle, when the vehicle is involved in a traffic accident, all the individuals in the vehicle are affected. This fact attracts the interest of researchers in investigating the major reasons of road incidents and developing various tools based on image processing and machine learning. These tools not only contribute to study and analyze the behavior of drivers in order to provide a report on the patterns of drivers on the roads but also help to mitigate major road-side accidents that otherwise results in the loss of economy and precious lives.

There are economic losses, apart from health damages and sometimes loss of life, due to road-side accidents. Authors in [6] estimated that the annual financial damages caused du to vehicle accidents in the United States in the year 2000 were USD 230.6 billion, and the total economic cost to society was

DOI: 10.14738/tmlai.85.9190
Publication Date: 01st Nov, 2020
URL: http://dx.doi.org/10.14738/tmlai.85.9190

approximately USD 493.3 billion. The direct cost of a single car accident was estimated to be approximately USD 14,000, of which USD 3,600 is the cost of damage to vehicles and other property. Therefore, the development of tools that detect behavior patterns of drivers that are able to reduce the number of accidents is very interesting area of research and has a high potential for development, since these methods generate great benefits for the society.

Authors in [7] present a study of a vehicle motion detection and tracking system through the monitoring of cameras, in order to control intersections. The study shows that there are cameras installed on the roads and other cameras mounted on vehicles. These cameras have their limitations. For instance, cameras can be sensitive to shadows that prevent a clean recording. In addition, climate conditions also interfere with the recording. Therefore, there is much in this filed which remains to be done.

Most accidents on the roads are caused by the imprudence of a driver; whether due to over-speeding, violation of some traffic signal among other factors. These types of events have been researched in [8]. The study provides the components of a driver assistance tool, which records the behavior of the driver when driving and allows him to have feedback on the infractions that may have been committed while driving. The common goal of hardware development that serves to detect infractions is to provide drivers with tools that allow them to be more responsible on the roads and thereby minimize accidents and avoid the violations of existing traffic laws.

The use of data loggers in a vehicle is very helpful. There is interesting research in this regard presented in [9]. The authors propose a detailed review of the benefits that these recorders can bring. The authors focus on the implementation of an IVDR using the drive diagnostics system and report the advantages of using these tools.

Within the framework of this research, we are oriented to see the usefulness of image processing-based algorithms to process images in order to be able to study their behavior and identify patterns on how the drivers respect the traffic signals and have a responsible and appropriate behavior on the roads.

2 Related Works

The implementation of image processing algorithms has been very useful in several areas. These techniques assist in analyzing various types of data, such as text, images, acoustic data. Based on these analysis, one can detect patterns and trends that are useful for decision making in their respective areas. Various algorithms have been proposed for finding similarity between two images. The most common algorithms for feature detection are Harris corner detector [10], scale invariant feature transform (SIFT) [11], speed up robust feature (SURF) [12] and oriented FAST and rotated BRIEF (ORB) [13].

Authors in [14] present an algorithm for the detection of moving vehicles making significant contributions to this issue. They conclude that currently many algorithms perform detections of moving objects by subtracting images (including a sequence of subtraction and subtraction of background images). The authors presented that the algorithm proposed in their work has high precision and they used the detection based on the optical flow of the pyramid through Experiment in Real AVI (audio video interleaved). Following presented are the main techniques used for detecting and monitoring vehicles.

Copyright © Society for Science and Education United Kingdom

2.1 Monitoring Methods Based on the Region

In these methods, the regions (bounding boxes) of moving objects are tracked to track vehicles [15]. The model uses the knowledge of the position and speed of each vehicle. In addition, this model uses a series of traffic scenes recorded by a stable camera for automobile monocular images. An evaluation of the traffic criteria, such as the numbering and classification of the vehicles that imply the suggested traffic observation scheme, has been suggested by [16]. The ratio of characteristics and density to classify the vehicles, in addition to the geometric features was used in order to eliminate false regions.

2.2 Monitoring Methods Based on 3D Models

The authors in [17] presented an occlusion detection approach based on a deformable model. A cube shape with six vertices is used. This cube is used to adapt to different vehicle types and sizes by changing the vertices for a better fit. Therefore, vehicle detection, segmentation and tracking can be achieved in an efficient manner.

2.3 Tracking methods based on characteristics

These methods are iterative and distinguishable frameworks based on edge points and modified SIFT descriptors [11]. The proposed framework shows a good performance for the classification of vehicles in surveillance videos despite the important challenges, such as the limitation of size and image quality and large intraclass differences.

A linearity characteristic technique is a line-based shading method that uses groups of lines to eliminate all undesirable tones. This method performs well on the occlusion resulting from shadows. This method represents an automatic traffic sorting and classification traffic observation system. Therefore, the particular vehicles are detected, segmented and tracked in a sequence of images by assembling, grouping and approximating of the 3D coordinates of the characteristic points of the vehicle.

2.4 Color and Tracking Methods Based on Patterns

A color analysis of the series of images of traffic is presented in [18]. The authors use the YCrCb color space for preliminary background construction and foreground segmentation. The location of the vehicle, vehicle tracking, shadow removal and background update algorithms are proposed. This system is able to perform under various climatic situations and is insensitive to illumination. A system for real-time traffic monitoring and continuous visual tracking of vehicles for the busy multi-lane motorway scene is presented in [19]. In this work, the authors use orthographic approximations for the pairing process. Tracking objects in visual surveillance systems is a challenging task for researchers [20]. To track the physical appearance of moving objects, such as vehicles, and identify them in a dynamic scene, we needed to locate the position, estimate the movement of these bubbles and follow these movements between two consecutive frames in the video scene as proposed by [21].

3 Methodology

In this work, we present a hybrid technique to detect the traffic infractions committed by the drivers. The problem we focus is a two-class problem, i.e., designing such an algorithm that is able to automatically detect those vehicles that are violating any traffic rule and causing issues to on-going traffic, that may result in any catastrophic accident. Therefore, the target is to detect the vehicles which have an anomalous driving pattern in comparison to other vehicles on the road. In addition, we also verify if those

patterns represent some sort of traffic violation. With the help of data analysis, we will also estimate the most frequent traffic violations committed by the drivers. The proposed techniques are applicable to the imagery data and video data.

The most common traffic violations are:

1. Illegal Parking. For instance, vehicles parked on bridges, tunnels, narrow areas of the road, level crossings, overpasses in grade changes, slopes and railroad crossings. This also includes the vehicles that are unloading or loading shipments in non-designated places.

2. Circulate in the opposite direction to authorized transit.

3. Cross an intersection or turn, the traffic light being red and there is no indication to the contrary.

4. Not respecting the established maximum or minimum speed limits or stopping the vehicle abruptly without any reason.

5. Not turning-on the indicators, or take precautions for changing lanes, taking U-turns and moving to the road shoulder for stopping without providing the indications.

3.1 Datasets

In order to implement the proposed work, we have used three datasets. The first dataset is presented in [22]. This dataset is prepared by the joint German project named Ko-PER. The data acquisition that include the dynamic models of on-going traffic in Ko-PER project is accomplished by deploying multiple laser-scanners and video cameras. All the raw data and the related calibration parameters are provided in ". mat" files that can be easily processed in MATLAB. This dataset contains raw laser sensor data, undistorted camera images and referenced data for selected vehicles. The data is illustrated in Fig. 1.

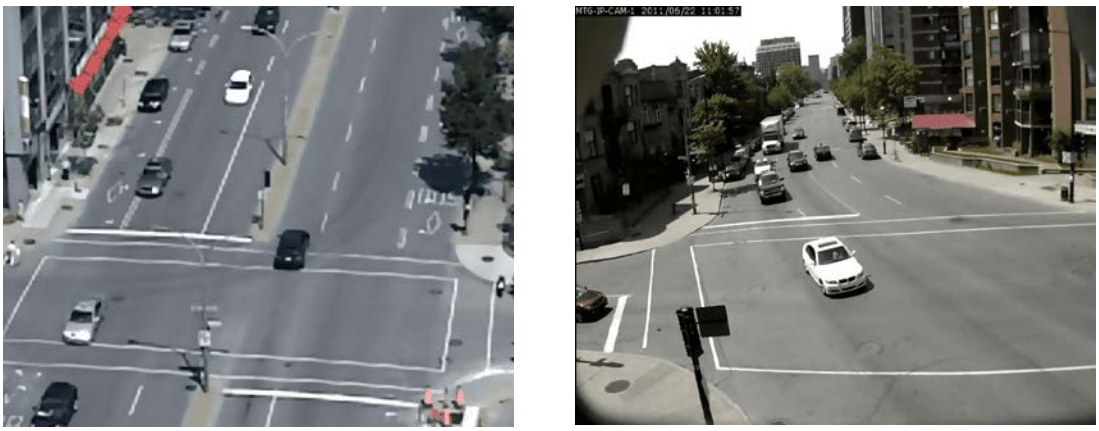


Figure 1. Sample frames from video data of road traffic presented in [22].

The second video dataset comprises of highway traffic [23][24]. The video data was collected over two days using a stationary camera overlooking I-5 in Seattle, WA. The video data was manually labelled to 3 different categories. These labels include light, medium, and heavy traffic, which correspond to free-flowing traffic, low speed traffic and very slow or stagnant vehicles, respectively. The video is provided by the courtesy of Washington State Department of Transportation. The illustration frames of this data are presented in Fig. 2.

Copyright © Society for Science and Education United Kingdom



Figure 2. Sample frames from video data of traffic on highway.

The third data set is presented in [25]. This dataset provides the videos of vehicles on highways captured using a pole-mounted stationary camera. The ground truth bounding box for each vehicle and their respective speed is also presented. We use this data set to estimate the speed of the vehicle using frame rate, in addition to other information. The errors of speed estimation are presented in the results section.

3.2 Interfering Factors in Data Acquisition

One of the main obstacles in this research work is the video recordings and imagery collection. Many factors significantly affect the collection of vehicle data. For example, the low luminosity, shadows, cloudiness, heavy rain, among other climatic factors often hinder the performance of the cameras in recordings. However, mitigating these issues is out of scope for this work.

3.3 Data Pre-processing

In order to remove the unnecessary details and noise from the images and video, we perform smoothing. The smoothing is performed using the Gaussian filter with a size of 5×5 pixels. The Gaussian filter is defined in (1)

$$G(x,y) = \frac{1}{2\pi\sigma^2}\, e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{1}$$

Where, x and y are the spatial coordinates of the image, and σ represents the standard deviation. An example of 3×3 Gaussian filter is presented in Fig. 3.

$$\frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Figure 3. A 3×3 Gaussian filter

3.4 Proposed Techniques

All the method proposed in this work are based on traditional methods. We present a hybrid technique that combines these methods to efficiently detect the traffic violations.

3.4.1 Lane Detection

Before detecting any infractions committed by the drivers, we first correctly detect the lanes in the video to check whether the vehicles are going in straight lane. The process of lane detection is done by applying the Hough transform [26].

First, we perform framing of the data, i.e., the video data is broken in the chunks, where interval of each individual chunk is of 10 milliseconds. After framing the data, we apply canny edge detection to extract the edges of the required objects, which in our case are lanes. For finding the intensity gradients, we apply Sobel filter to obtain the edges [27]. The Sobel matrices are defined as

$$H_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad H_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Therefore, the Sobel operation is defined as

$$H = \sqrt{H_x^2 + H_y^2} \tag{2}$$

$$\Theta = \operatorname{atan}\left(\frac{H_y}{H_x}\right) \tag{3}$$

Where, the H and Θ represent the gradient magnitude and gradient direction respectively. H_x and H_y represent $\frac{\partial I}{\partial x}$ and $\frac{\partial I}{\partial y}$, respectively and I is the blurred image. After obtaining the edges, we apply non-maximal suppression to obtain the thickest edges. This process also helps in the removal of false edges. A mask is then used to extract the boundary of lane; this mask depends on the area where camera is present. In our case, a triangular mask was used. Lastly, Hough transform is used to detect the left and right boundary line of lane [26]. This was done by taking a threshold number of intersecting points in Hough space which lie on same line in cartesian space. The mathematical representation of Hough transform is

$$r = x\cos\theta + y\sin\theta \tag{4}$$

Where r represents the length of the normal from the origin of cartesian space to the line and θ represents the orientation of the line represented by r with respect to X-axis of cartesian space. Therefore, each point in cartesian space is represented as a line in the Hough space.

3.4.2 Speed Calculation

The first step of speed calculation is the detection of vehicles in the video data. We use Haar Cascade method for vehicle detection in the video data [28]. The Haar cascade method is extremely efficient in terms of detections and is very fast as well. The Haar Cacade algorithm has three key features. First, a new technique for representing an image called integral image, which represents the intermediate image between original image and box features. At each location of the integral image, the value is represented by the sum of all the pixels situated above and towards left of the spatial location (x, y). Please note that the calculated sum is inclusive of the pixel value located at (x, y). The transformation of input frame to integral images allows fast feature computation. Integral image is defined as

Copyright © Society for Science and Education United Kingdom

$$II(x,y) = \sum_{x' \le x, y' \le y} I(x',y') \tag{5}$$

Where, $II(x,y)$ and $I(x,y)$ represent the integral image and the input image, respectively. The transformation to integral image is accomplished by set of equations [28]

$$s(x,y) = s(x,y-1) + I(x,y) \tag{6}$$

$$II(x,y) = II(x-1,y) + s(x,y) \tag{7}$$

Where, $s(x,y)$ represents cumulative row sum, $s(x,-1) = 0$ and $II(-1,y) = 0$.

Second, after the integral representation is constructed, AdaBoost based learning algorithm is applied for learning [29]. This learning considers a small set of most efficient features. The learning performed using AdaBoost performs better as compared to other techniques, such as steerable filters. Third, the idea of modified AdaBoost is to design an algorithm which comprises of small and efficient boosted classifiers. This new algorithm has the capability to reject false positive while maintaining the quantity of true positives.

We train our algorithm for detecting the vehicles on the road using the aforementioned data. Our algorithm then provides the bounding boxes for the true positives, i.e., vehicles, which can now be tracked for speed estimation. Each bounding box that is containing vehicle is assigned a unique id. The tracking of the bounded vehicles is performed using correlation tracker. The correlation trackers take their base from the well-known Fourier transform, as the correlation tracker processes in the frequency domain. Consider that the Fourier transform of the input image is $T = F(I(x,y))$, where, T is the Fourier transform of the input image and $T' = F(G(x,y))$ is the Fourier transform of the filter or specified region, where $G(x,y)$ is the filter or region of an image under consideration. Correlation can now be defined as

$$H = T.T' \tag{8}$$

Please note that when the signals are convolved in the time domain, they are simply multiplied in the frequency domain, as put forward by the Fourier analysis. However, the output of such correlation filter has to be transformed back to the time/spatial domain. This is a serious bottle neck in correlation filter's design.

For our work, we make use of minimum output sum of squared error (MOSSE) filter for correlation-based tracking [30]. The MOSSE filter is efficient as it requires fewer images for training. For training process, MOSSE filter requires set of input images I_i and a set of output images I_o. In our case, the I_i contains vehicles images extracted as a ground truth. In terms of Fourier transform the relation between these entities become

$$T_i' = \frac{H_i}{T_i} \tag{9}$$

Where, T_i', H_i and T_i represent the Fourier transform of input data, correlation and input images. Now, we implement a convex optimization solution in such a way that the filter T_i' minimizes the error of input and desired output.

$$minimize\ (T_i') \sum_i |T_i . T_i' - H_i|^2 \tag{10}$$

Taking the derivative and putting it equal to zero to find the minima of the quadratic error function, the closed form expression for the MOSSE filter is

$$T_i' = \frac{\sum_i H_i . T_i}{\sum_i T_i . T_i} \tag{11}$$

This is the mathematical form of our correlation filter. It is notable that, the previously proposed correlation filters in literature require centric objects. Contrary, the objects in MOSSE filter need not be cantered because the object is approximated using Gaussian density function, and even small part of the object will suffice for tracking it. Please note that the Gaussian density is considered with $\sigma = 2$. However, in some cases where object comprises of large windows, MOSSE detector may fail to track the object. In our work we use the correlation tracker presented in [35]. The authors present a correlation filter based on MOSSE filter that is able to efficiently estimate the location of the object in the future frames.

The calculation of vehicle's speed using the pixels and frame information is an interesting research problem. Various techniques have been proposed to efficiently estimate the speed of vehicle using video data [31-33]. Authors in [34] propose a new technique for estimating speed of vehicles. The authors consider three frames simultaneously, i.e., $I_n(x,y)$, $I_{n-1}(x,y)$ and $I_{n-2}(x,y)$ (where these frame represent the n, $n-1$ and $n-2$ represent the current time and the previous time, respectively) for estimating, if the object under consideration is moving. The vehicle is moving if the following set of equations is satisfied.

$$I_n(x,y) - I_{n-1}(x,y) > T_n(x,y) \tag{12}$$

$$I_n(x,y) - I_{n-2}(x,y) > T'_n(x,y) \tag{13}$$

Where, $T_n(x,y)$ and $T_n'(x,y)$ represent the threshold.

The next step is the suppression of image background. This is accomplished by

$$I_n'(x,y) = I_n(x,y) - B_n(x,y) \tag{14}$$

Where $I_n'(x,y)$ and $B_n(x,y)$ represent the image without background and the background image, respectively. The difference between the background image and the objects of interest is done by using AdaBoost classifier, which provides the vehicles boundaries. The rest of the image is regarded as a background. The distance can now be calculated by

$$d = \sqrt{(x_{n-1} - x_n)^2 + (y_{n-1} - y_n)^2} \tag{15}$$

Where, (x_n, y_n) and (x_{n-1}, y_{n-1}) represent the centroid of the vehicle in the n^{th} frame and $n^{th} - 1$ frame, respectively. Therefore, we can calculate the speed of the vehicle as

$$s = \frac{h \times d}{I_n(x,y) - I_{n-1}(x,y) \times fps} \tag{16}$$

Copyright © Society for Science and Education United Kingdom

Where, h and fps represent the height of the vehicle and frame rate of video, respectively. This algorithm is able to identify those vehicles which are travelling below or above the allowed speed on any road or a free-way. Moreover, we can also detect if any vehicle is parked at an inappropriate place or if a fast-moving vehicle stopped suddenly. The same algorithm can be used to detect if the vehicle was travelling in the wrong lane (vehicle moving in opposite direction).

4 Results and Analysis

4.1 Simulation Setup

In this section, we present our simulation environment that was used to detect the lanes and compute vehicle's speed for monitoring and detecting the driving patterns. All the computations are performed using python. The python libraries used for the computations include numpy, OpenCV, dlib, time, threading and math.

4.1.1 Lane Detection

For smoothing, we used a Gaussian filter of 5×5 with $\sqrt{\sigma} = 1$. The canny edge detection was performed on the blurred image using Sobel filter. The values for lower and upper threshold for hysteresis were 50 and 150, respectively. The triangular masked used for boundary extraction was created using the following python code

```
Import numpy as np

Height = frame.shape[0]

Triangular mask = np.array([ [(0, height), (800, height), (380, 290)] ])
```

For calculating the Hough transform, we considered rho=2, $\theta = \pi/180$, threshold = 100, minimum line length=100 and minimum line gap=50.

4.1.2 Speed Calculation

We use the Haar Cascade trained model for cars and buses provided in OpenCV. The vehicles in the video data are detected using 'CascadeClassifier()' method. Once the vehicle is detected in the video, it is tracked by making use of correlation filter provided in the dlib library by using the 'dlib.correlation_tracker()' and CascadeClassifier.detectMultiScale(image, scaleFactor=1.1, minNeighbors=13, minSize=18, maxSize=(24,24))' methods. The input arguments of this correlation tracker are the coordinates of the vehicle detected in the previous phase of the algorithm. For the purpose of speed calculation, fps is set at 18.

4.2 Results

We use the vehicle detection and vehicle accuracy to analyze our proposed method. The results for detection and few traffic patterns are presented in Fig. 4. The Table 1 summarizes the accuracy of vehicle detections in terms of a confusion matrix. Please note that, as the confusion matrix is made for vehicle detection, all the vehicles form one class and all the other objects, and the background is considered as the other class. However, we don't focus on the background detections unless background or other objects are classified as a vehicle.

Table 1: The average confusion matrix for vehicle detection in the video data.

True Positive = 96%	False Positive = 2%
False Negative = 2%	True Negative = 0% (not focused in computations)

Table 2: Approximate error in the vehicle speed estimation.

Actual Speed	Estimated Speed	Approximate Error
10 km/hr	8.5 - 11.5km/hr	±15%
20km/hr	19 - 21.5km/hr	±6.5%
30km/hr	27.5 - 31.5km/hr	±7%
40km/hr	37 – 43.5km/hr	±9%
50km/hr	46 - 53km/hr	±7.5%
80km/hr	74.5 – 87km/hr	±8%
100km/hr	92 - 106km/hr	±7%

The algorithm is also able to correctly estimate the speed of the moving vehicles with an approximate error presented in Table 2. We use mean square error (MSE) to compute the error in vehicle's speed.

$$MSE = \frac{1}{n}\sum_{i=1}^{n} Y_i - Y_i' \tag{17}$$

Where Y_i, Y_i' and n represent the actual speed, estimated speed and the total number of vehicles used for speed estimation, respectively.



Figure 4. (From left to right) (a)(b)(c) Over speeding vehicle detection. (d) Vehicle parked on a highway. (e) Vehicle trying to take wrong U-Turn.

Copyright © Society for Science and Education United Kingdom

4.3 Analysis

As evident from the results, we were able to correctly detect the vehicles and estimate the speed of each vehicle. In addition, by making use of the lane detection, we can identify the vehicle which is continuously changing lanes and has a high speed. The speed estimation algorithms not only enable us to estimate speed, in addition, it allows us to identify the vehicles that are taking wrong U-turns, stopping at an inappropriate place and moving in opposite direction in the wrong lane. For example, if the vehicle speed is estimated to be 0km/h in 100 consecutive frames then the vehicle is assumed as a parked vehicle.

5 Conclusion

In this work, we presented a hybrid technique for automated monitoring and detecting the traffic infraction patterns. The automated process helps in the reduction of traffic violations and reduce road incidents. We proposed two major methods including the lane detection and speed estimation using video data. The proposed method helps in the identification of those vehicles that show violation patterns, such as wrong U-turns, inappropriate parking, moving at a high or low speed than the allowed speed limit. We use Hough transform and Canny edge detection for lane detection in real time. Similarly, we use Haar Cascade for detecting the vehicles in the video data. Once the vehicles are detected, we use a correlation filter to track them. Similarly, in addition to tracking, the speed of the tracked vehicle is computed using the frame rate and correlation filter. The results presented in this work makes it evident that the proposed method is successfully able to assist the authorities in the indication of those vehicles that are in violations of traffic regulations. Thus, allowing the traffic to flow freely and significantly reduce the number of road accidents. The results also reveal that the proposed method is able to detect the vehicles with a precision of 95%. Similarly, the proposed work is also able to correctly predict the vehicle's speed with an error of ±15% at low speed and ±7% at high speed.

REFERENCES

[1] Uy AC, Bedruz RA, Quiros AR, Bandala A, Dadios EP. Machine vision for traffic violation detection system through genetic algorithm. In 2015 International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM) 2015 Dec 9 (pp. 1-7). IEEE.

[2] Klubsuwan K, Koodtalang W, Mungsing S. Traffic violation detection using multiple trajectories evaluation of vehicles. In 2013 4th International Conference on Intelligent Systems, Modelling and Simulation 2013 Jan 29 (pp. 220-224). IEEE.

[3] Gomes SL, Rebouças ED, Neto EC, Papa JP, de Albuquerque VH, Rebouças Filho PP, Tavares JM. Embedded real-time speed limit sign recognition using image processing and machine learning techniques. Neural Computing and Applications. 2017 Dec 1;28(1):573-84.

[4] Zhang L, Yang F, Zhang YD, Zhu YJ. Road crack detection using deep convolutional neural network. In 2016 IEEE international conference on image processing (ICIP) 2016 Sep 25 (pp. 3708-3712). IEEE.

[5] Xu B, Loce RP. A machine learning approach for detecting cell phone usage. In Video Surveillance and Transportation Imaging Applications 2015 Mar 4 (Vol. 9407, p. 94070A). International Society for Optics and Photonics.

[6] Blincoe LJ. The Economic Cost of Motor Vehicle Crashes, 2000, May. Washington, DC: US Department of Transportation, National Highway Traffic Safety Administration, DOT HS.;809:446.

[7] Datondji SR, Dupuis Y, Subirats P, Vasseur P. A survey of vision-based traffic monitoring of road intersections. IEEE transactions on intelligent transportation systems. 2016 Apr 22;17(10):2681-98.

[8] Aliane N, Fernandez J, Mata M, Bemposta S. A system for traffic violation detection. Sensors. 2014;14(11):22113-27.

[9] [9] Toledo T, Musicant O, Lotan T. In-vehicle data recorders for monitoring and feedback on drivers' behavior. Transportation Research Part C: Emerging Technologies. 2008 Jun 1;16(3):320-31.

[10] Harris CG, Stephens M. A combined corner and edge detector. InAlvey vision conference 1988 Aug 31 (Vol. 15, No. 50, pp. 10-5244).

[11] Lowe DG. Distinctive image features from scale-invariant keypoints. International journal of computer vision. 2004 Nov 1;60(2):91-110.

[12] Bay H, Tuytelaars T, Van Gool L. Surf: Speeded up robust features. InEuropean conference on computer vision 2006 May 7 (pp. 404-417). Springer, Berlin, Heidelberg.

[13] Rublee E, Rabaud V, Konolige K, Bradski GR. ORB: An efficient alternative to SIFT or SURF. InICCV 2011 Nov 6 (Vol. 11, No. 1, p. 2).

[14] Zhan W, Ji X. Algorithm research on moving vehicles detection. Procedia Engineering. 2011 Jan 1;15:5483-7.

[15] Gupte S, Masoud O, Martin RF, Papanikolopoulos NP. Detection and classification of vehicles. IEEE Transactions on intelligent transportation systems. 2002 Aug 7;3(1):37-47.

[16] Lai JC, Huang SS, Tseng CC. Image-based vehicle tracking and classification on the highway. In The 2010 International Conference on Green Circuits and Systems 2010 Jun 21 (pp. 666-670). IEEE.

[17] Yung NH, Lai AH. Detection of vehicle occlusion using a generalized deformable model. In ISCAS'98. Proceedings of the 1998 IEEE International Symposium on Circuits and Systems (Cat. No. 98CH36187) 1998 May 31 (Vol. 4, pp. 154-157). IEEE.

[18] Huang MC, Yen SH. A real-time and color-based computer vision for traffic monitoring system. In 2004 IEEE International Conference on Multimedia and Expo (ICME) (IEEE Cat. No. 04TH8763) 2004 Jun 27 (Vol. 3, pp. 2119-2122). IEEE.

[19] Sullivan GD, Baker KD, Worrall AD, Attwood CI, Remagnino PM. Model-based vehicle detection and classification using orthographic approximations. Image and vision computing. 1997 Aug 1;15(8):649-54.

[20] Shan C, Porikli F, Xiang T, Gong S, editors. Video Analytics for Business Intelligence. Springer Berlin Heidelberg; 2012 Apr 5.

Copyright © Society for Science and Education United Kingdom

[21] Rhee SH, Han S, koo Kim P, Ahmad MB, Park JA. Vehicle tracking using image processing techniques. In International Conference on Rough Sets and Current Trends in Computing 2004 Jun 1 (pp. 671-678). Springer, Berlin, Heidelberg.

[22] Strigel E, Meissner D, Seeliger F, Wilking B, Dietmayer K. The ko-per intersection laser scanner and video dataset. In 17th International IEEE Conference on Intelligent Transportation Systems (ITSC) 2014 Oct 8 (pp. 1900-1901). IEEE.

[23] Chan AB, Vasconcelos N. Probabilistic kernels for the classification of auto-regressive visual processes. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) 2005 Jun 20 (Vol. 1, pp. 846-851). IEEE.

[24] Chan AB, Vasconcelos N. Classification and retrieval of traffic video using auto-regressive stochastic processes. In IEEE Proceedings. Intelligent Vehicles Symposium, 2005. 2005 Jun 6 (pp. 771-776). IEEE.

[25] Sochor J, Juránek R, Špaňhel J, Maršík L, Široký A, Herout A, Zemčík P. Comprehensive data set for automatic single camera visual speed measurement. IEEE Transactions on Intelligent Transportation Systems. 2018 May 8;20(5):1633-43.

[26] Duda RO, Hart PE. Use of the Hough transformation to detect lines and curves in pictures. Sri International Menlo Park Ca Artificial Intelligence Center; 1971 Apr.

[27] Kanopoulos N, Vasanthavada N, Baker RL. Design of an image edge detection filter using the Sobel operator. IEEE Journal of solid-state circuits. 1988 Apr;23(2):358-67.

[28] Viola P, Jones M. Rapid object detection using a boosted cascade of simple features. CVPR (1). 2001 Dec 8;1(511-518):3.

[29] Schapire RE, Freund Y. A decision-theoretic generalization of on-line learning and an application to boosting. In Second European Conference on Computational Learning Theory 1995 Mar 30 (pp. 23-37).

[30] Bolme DS, Beveridge JR, Draper BA, Lui YM. Visual object tracking using adaptive correlation filters. In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2010 Jun 13 (pp. 2544-2550). IEEE.

[31] Javadi S, Dahl M, Pettersson MI. Vehicle speed measurement model for video-based systems. Computers & Electrical Engineering. 2019 Jun 1;76:238-48..

[32] Lee J, Roh S, Shin J, Sohn K. Image-based learning to measure the space mean speed on a stretch of road without the need to tag images with labels. Sensors. 2019 Jan;19(5):1227.

[33] Song W. Image-Based Roadway Assessment Using Convolutional Neural Networks.

[34] Moazzam M, Haque M, Uddin M. Image-Based Vehicle Speed Estimation. Journal of Computer and Communications, 7, 1-5. doi: 10.4236/jcc.2019.76001.

[35] Danelljan M, Häger G, Khan F, Felsberg M. Accurate scale estimation for robust visual tracking. InBritish Machine Vision Conference, Nottingham, September 1-5, 2014 2014. BMVA Press.