



Vehicle speed and dimensions estimation using on-road cameras by identifying popular vehicles

H. Khosravi*, R. Asgarian Dehkordi, and A. Ahmadyfard

Faculty of Electrical Engineering, Shahrood University of Technology, Shahrood, Iran.

Received 28 January 2020; received in revised form 20 October 2020; accepted 16 November 2020

KEYWORDS

Camera calibration;
 Speed estimation;
 Dimension estimation;
 Bag of word;
 Vehicle recognition.

Abstract. One of the major issues related to traffic monitoring systems is analysis of the behavior of vehicles and identification of their characteristics. In this paper, an automated algorithm is proposed for calibration of road cameras. This calibration is used to estimate the speed and dimensions of passing vehicles. By using this method, a motion plane is obtained in the initial frames according to the direction of moving vehicles. After modeling the background, removing shadows, and identifying the exact boundaries of passing vehicles, a number of common vehicles are recognized through the bag of words method. Given the actual metric dimensions of these vehicles and the equivalent dimensions on the motion plane, the metric coefficients are calculated and the calibration process is completed. Each passing vehicle is projected on the motion plane, and speed and dimensions are calculated by tracking it along the road. To test the accuracy of the proposed method, a ground-truth video dataset is constructed by simultaneously capturing road vehicles using RGB and speed cameras. Furthermore, to identify common vehicles, a dataset of vehicle images is collected. The proposed method is evaluated based on our dataset and the mean error of 1.15 km/h is achieved. Accordingly, it outperforms previous methods.

© 2022 Sharif University of Technology. All rights reserved.

1. Introduction

In recent years, the issue of the use of video surveillance cameras to analyze traffic and obtain some information about the running vehicles remains a popular topic in the field of machine vision and intelligent transportation system. In this context, relevant approaches include vehicle speed estimation [1,2], dimensions approximation [3], make-and-model recognition [4,5], and orientation estimation [6]. The main challenge in estimating these parameters is camera calibration. Calibration is the process of estimating the internal and external parameters of the camera. Internal parameters

include focal length (f_x, f_y) , center point of the image (P_x, P_y) , and radial distortion of the camera. External parameters include translation vector T and rotation matrix R with respect to the reference point in the 3D space. Calibration helps tackle the issue and allows measuring the dimensions of objects on the scene [7].

In most of the calibration methods presented so far, the user must consider some parameters associated with the scene or camera. It is implied that these methods must be fine-tuned for each road separately. In addition, most of these methods only estimate the speed of vehicles [8–11]. To ensure a more convenient and accurate analysis, a method that is fully automated and provides more information about vehicles is required.

Automated calibration methods presented so far function based on certain information including the size of the vehicle license plate [12], average vehicle dimension as a set of parallel lane markings [13], or

*. Corresponding author.

E-mail address: hosseinkhosravi@shahroodut.ac.ir (H. Khosravi)

dimension of the dominant vehicle [14]. Although these are good techniques, they are subject to some drawbacks. For example, any error in license plate localization increases the calibration error. Finding the average dimension of vehicles requires a considerable amount of time for an acceptable number of vehicles passing in front of the camera. In addition, it may encounter errors in dimension estimation that reduce the overall performance. Finally, if the diversity of vehicles in a country is high, identifying the dominant vehicle is not an easy task.

The main contribution of this paper is to provide a fully automated algorithm to find the dimensions and speeds of the vehicles. Detecting the speed of a passing vehicle is a major requirement for the law enforcement systems and vehicle dimensions are important in identifying the class and type of vehicle and analyzing the amount of road traffic [15]. In the proposed method, several types of common cars are identified in the initial frames of the video. Then, according to the dimensions of these vehicles, the calibration is accomplished. To identify common cars, it is necessary to provide a dataset of vehicle images and train a classifier on them. To this end, a dataset was collected from taking pictures of on-road vehicles. To evaluate the accuracy of the proposed method, a ground-truth video dataset was also constructed through simultaneous capturing of the road vehicles using RGB and speed cameras.

The rest of this study is organized as follows. Related works are reviewed in the next section. Section 3 describes the proposed method. Section 4 compares the performance of the proposed algorithm with those of other methods. Finally, Section 5 draws the paper to conclusions.

2. Related works

The methods related to the proposed algorithm are grouped into two categories: calibration methods and vehicle-type identification methods.

2.1. Calibration and shadow removal methods

You and Zheng [16] used two vanishing points to calibrate the camera. The first vanishing point is in the direction of moving vehicles, while the second point is in another direction perpendicular to the road lines. In this method, the scale is calculated according to the height of the camera from the road surface. In [17], after identifying the license plate for several vehicles in different locations on the road, the size of the license plate was used for calibration. Luvizon et al. [18] proposed a speed estimation algorithm based on license plate detection and tracking. In this method, at first, the vehicle is identified. Then, the vehicle speed is determined by finding the location of the license plate and tracking it in consecutive frames.

In another study [19], vehicle lights were detected and tracked to count the number of vehicles and obtain their speeds at night. This method requires the angle and the height of the camera with respect to the road for camera calibration. In the case of another method presented by Eslami et al. [14], some vehicles were first identified in consecutive frames. Then, the location of the license plate and its characters were extracted and the calibration parameters of the camera were determined according to the standard dimensions of the license plate in Iran.

Famouri et al. [20] proposed a method that consisted of two general phases. At the first phase, a motion plane was determined by identifying the license plates and tracking the central point of them. At the second phase, the vehicle speed was computed by tracking and projecting the license plates on this plane. They used an object detector for license plate localization which was trained with about 4,000 license plates of Iranian cars.

Khosravi and Aghayan [21] presented a real-time method for estimating the speed of passing vehicles. In this method, a rectangular part of the road is manually selected and considered as the place of entrance and departure of vehicles. To remove perspectives from the selected region, the projective transformation is used. The metric coefficient in this method is computed according to the dimensions of the selected region.

Dubská et al. [22] presented an automatic camera calibration for traffic understanding. In this method, the coordinates of the vanishing points are computed in the initial frames. Then, the vehicle bounding box is formed by detecting moving vehicles and using camera parameters. By identifying a few cars and projecting the car boxes on the hypothetical ground plane, the scale factor is calculated. Projection of the car box on a hypothetical ground plane and the use of a scale factor allow this method to compute the metric dimensions and speed of the vehicles in each frame. The average error of this method in the speed and distance estimation is less than 2%.

Authors in [23], which is the previous work of the current authors, provided a fully automated method for estimating car dimensions. In this method, first, the calibration parameters are calculated. Then, the vehicle bounding box is formed and its dimensions are obtained in meters by applying transformation and using the scale coefficient. This method can create a high-precision 3D vehicle box and its average error is 1.5% in the dimension estimation. In this work, the dataset used is of good quality and has no challenges. Furthermore, in this method, calculation of the scale coefficient is time-consuming and sensitive to shadow and other optical variations of the environment.

One major challenge that can cause problems for vehicle dimension estimation and vehicle type recogni-

tion is the presence of shadow in the images. So far, shadow removal in the case of moving cars has been considered only in few methods. Lin et al. [24] utilized the edges of an image as a stable feature in different lighting conditions to remove shadows of the moving cars. In this method, assuming that the background is fixed, the exact boundary of the car is determined by analyzing the edges and colors of the background and foreground.

2.2. Vehicle identification methods

Wang et al. [25] developed a deep neural network for vehicle classification based on ALEXNET architecture [26]. It consists of five convolutional layers, three pooling layers, and three fully-connected layers. Faster Region proposal CNN (RCNN) [27] is used to localize the vehicles. They pointed out that if the training and test images were taken by different cameras, the error rate would increase. Thus, to reduce camera dependency, they used a large dataset of vehicles from web images.

Yang et al. [28] collected a large dataset to identify the exact type of vehicles and proposed a method using a Convolutional Neural Network (CNN) that can identify vehicles under different conditions. In [29], the authors used two CNN neural networks: the first and second networks were used to detect a vehicle and its type, respectively.

Gholamalinezhad and Khosravi [30] prepared a large dataset of Iranian vehicles, called IRVD. This dataset has 110,366 images of five classes captured in different weather and lighting conditions. They also developed a light-weight CNN composed of five convolutional layers for vehicle classification. Furthermore, batch normalization and data augmentation were

employed to improve the model accuracy. The final model achieved an accuracy rate of 99.07%, which is comparable with the results of ResNet50 architecture.

Hsieh et al. [31] proposed a symmetrical Speeded Up Robust Feature (SURF) for vehicle recognition. In this method, at first, the Scale Invariant Feature Transform (SIFT) key points are extracted for the entire image. These key points are then analyzed and according to the horizontal symmetry of the vehicles, they are localized. Finally, vehicle type is identified using Histogram of Oriented Gradients (HOG) and SURF descriptors and the Support Vector Machine (SVM) classifier.

Biglari et al. [4] presented a part-based approach that attempts to find distinctive parts of each subgroup of vehicles. They pointed out that the use of deep CNN networks required heavy computing, a lot of memory, and much time. Therefore, the mentioned authors used the HOG feature to describe each section and SVM for classification. They used a dataset of 5991 vehicle images of 28 classes taken from the frontal view.

3. The proposed method

Figure 1 shows a brief flowchart of the proposed method. It is composed of three general phases, in which the first and second phases are preprocessing stages for the third phase. In the first stage, vanishing points and focal length of the camera are determined according to the driving direction in the initial frames of the input video. These parameters are calculated only once at the beginning.

At the second phase, the shadow is removed and then, the exact 2D boundary of the car is determined

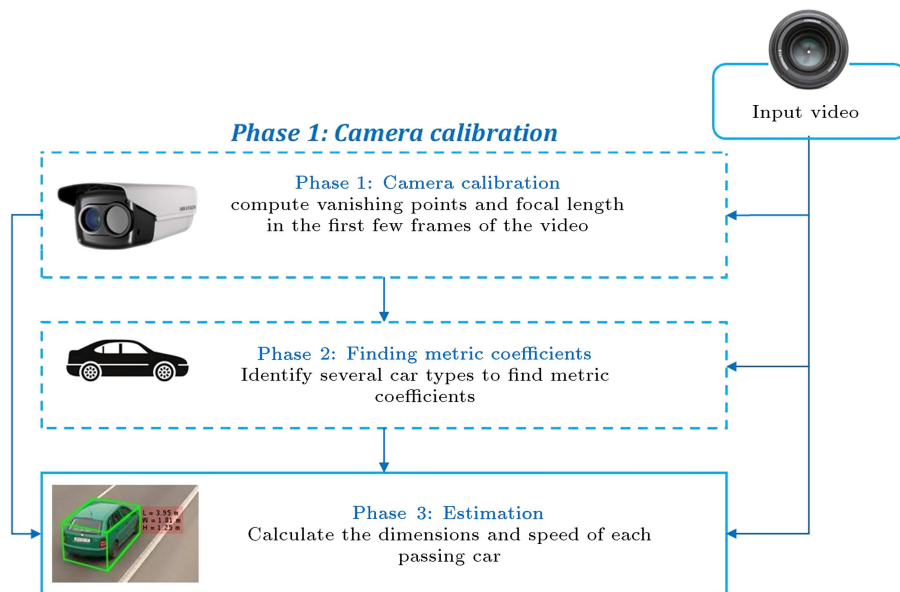


Figure 1. Three phases of the proposed method for speed and dimension estimation. Phases 1 and 2 are temporary.

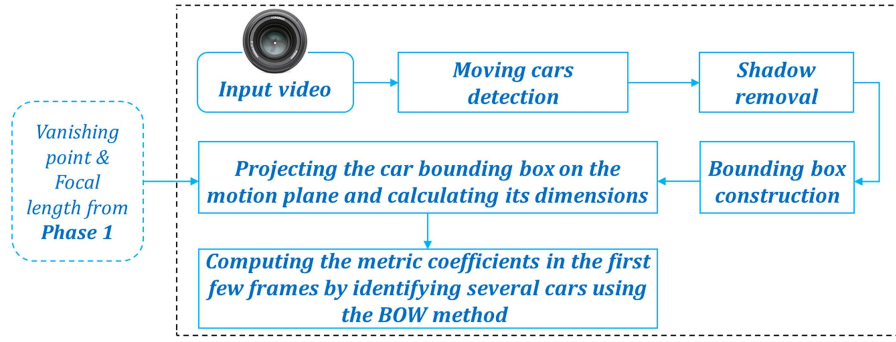


Figure 2. Flowchart of computing metric coefficients (Phase II).

and its 3D bounding box is constructed using the coordinates of the vanishing points. To remove the perspective, the coordinates of the vehicle box corners are projected on the motion plane and the length, width, and height of the vehicle are calculated in pixels. However, to express these parameters in meters, metric coefficients are required. Metric coefficients are obtained by identifying some popular vehicles. This is conducted using a Bag of Words (BoW) classifier. Details are given in Section 3.2.4.

After these two stages, the coordinates of the vanishing points and the metric coefficients are determined. Thus, it is possible to estimate the actual dimensions and speed of vehicles. At the third phase, cars are detected, the shadows are removed, and the bounding box coordinates on the motion plane are obtained. Then, by tracking each car, histograms are formed to indicate the length, width, height, and speed of each car. Finally, these parameters are precisely determined by analyzing the histograms and using the metric coefficients.

3.1. Phase I: Camera calibration

We used the method presented in [7] to estimate the coordinates of the vanishing points (VP_1 , VP_2 , and VP_3) and the focal length f . In this method, VP_1 and VP_2 are obtained based on the direction of moving cars, and then f is obtained from Eq. (1):

$$VP_1 = \begin{bmatrix} u_x \\ u_y \end{bmatrix}, \quad VP_2 = \begin{bmatrix} v_x \\ v_y \end{bmatrix}, \quad C = \begin{bmatrix} p_x \\ p_y \end{bmatrix},$$

$$f = \sqrt{-(VP_1 - C)^T (VP_2 - C)}, \quad (1)$$

where C is the center coordinate of the image. Based on f , the third vanishing point is obtained by Eq. (2):

$$V'P_1 = \begin{bmatrix} u_x \\ u_y \\ f \end{bmatrix}, \quad V'P_2 = \begin{bmatrix} v_x \\ v_y \\ f \end{bmatrix}, \quad c' = \begin{bmatrix} p_x \\ p_y \\ 0 \end{bmatrix},$$

$$V'P_3 = (V'P_1 - c') \times (V'P_2 - c'). \quad (2)$$

The first two components of $V'P_3$ are considered as the third vanishing point VP_3 . These parameters are

calculated using the first 2000 frames of the input video. Further details about the calculation of these parameters are given in [32].

3.2. Phase II: Computing metric coefficients

Figure 2 shows the flowchart of the second phase. Here, the metric coefficients are calculated through the identification of some pre-trained vehicles. Details of each step are described in the following.

3.2.1. Background modeling and shadow removal

Due to the movement of vehicles, we can use foreground detection methods to detect the boundaries of cars. Several methods have been proposed based on this approach. After evaluating different methods, the Zivkovic method [33] has been selected for background modeling. This method, which is an Improved version of the Gaussian Mixture Model (IGMM), is of high speed and good resistance against soft shadow and noise, but still has problems with hard shadow. Considering shadow as a part of the vehicle causes problems for constructing the bounding box of the car. In the proposed method, a shadow removal algorithm, proposed in [24], was employed. This method is fast and accurate and helps obtain the exact boundaries of the vehicles in the presence of soft and hard shadows.

3.2.2. Constructing 3D bounding box of the vehicle

To construct a bounding box for a vehicle, tangent lines are required for each vanishing point. To find these lines, edges of the vehicle are detected using the Canny edge detector. Then, lines are drawn from each edge to the vanishing points. According to these lines, tangent lines and bounding boxes are obtained for each vehicle. In the first row of Figure 3, the green lines are the tangents drawn to the first vanishing point; the red lines are tangents drawn to the second point; and the blue lines are the tangents drawn to the third point. The image in the second row of Figure 3 shows the 3D box of a vehicle. Points A , B , C , and E are four coordinates of the vehicle corners that are used to estimate the dimensions. The lengths of lines AE , AB , and AC are used as the height, width, and length of the vehicle.

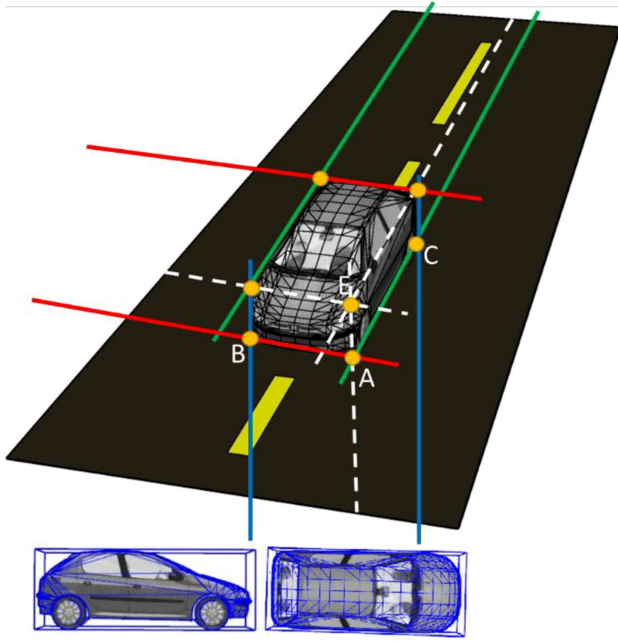


Figure 3. Constructing the 3D bounding box for a vehicle.

3.2.3. Projecting the bounding box on the hypothetical road surface and calculating the dimensions

As the vehicle moves along the first vanishing point, the size of the vehicle becomes smaller and the dimensions obtained for the vehicle change over time. To solve this problem, a motion plane is constructed that is parallel to the actual road, but has no perspective. By projecting the car onto the motion plane, its dimensions would not change significantly over time.

The motion plane ϕ should be parallel to the actual road. Thus, this plane is parallel to the axes aligned with the first and second vanishing points and its normal vector can be obtained from the cross product of these two vectors. To illustrate the road surface, we consider the center of the image plane, $P = [P_x P_y f]^T$, and the location of the camera, $O = [P_x P_y 0]^T$ (P_x and P_y are centers of the image). As mentioned before, the motion plane is parallel to the main road, but the scale factor is still unknown. The metric coefficients are employed to correct the obtained dimensions. Figure 4 shows the camera, image plane, and two motion planes with different distances to the camera. Considering the motion plane ϕ , the points A_w , B_w , and C_w are obtained by projecting the points A , B , and C from the image plane onto the plane ϕ and the point E_w is obtained by projecting the point E on the normal vector of ϕ (Eq. (3)):

$$\begin{aligned} A_w &= \phi \cap \overrightarrow{OA}, C_w = \phi \cap \overrightarrow{OC}, \\ B_w &= \phi \cap \overrightarrow{OB}, E_w = N \cap \overrightarrow{OE}. \end{aligned} \quad (3)$$

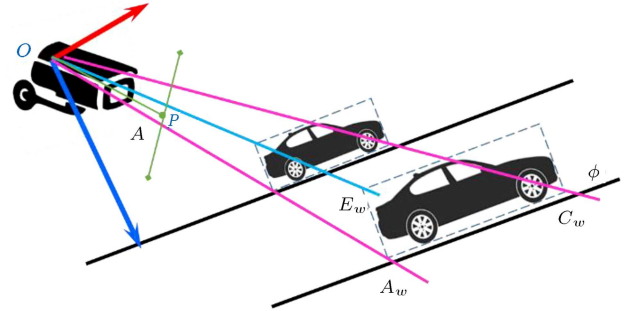


Figure 4. The camera, image plane, and desired motion planes.

After projecting the points, the length, width, and height of the vehicle are determined on the motion plane (ϕ) as follows:

$$L = |A_w C_w|, W = |A_w B_w|, H = |A_w E_w|. \quad (4)$$

It should be noted that these dimensions are not the actual metric dimensions. Using Eq. (4), we can estimate the dimensions of the vehicle in each frame for the desired plane. Upon comparing the values obtained for the dimensions of a particular vehicle, it is found that these values experience small variations in successive frames. These variations result from the errors in foreground detection and inaccurate construction of the 3D bounding box.

To have more exact dimensions, we considered tracking each vehicle in successive frames using optical flow [34] and constructing histograms of its length, width, and height. Now, the positions of the maximum values of these histograms can be selected as more accurate dimensions of the car.

3.2.4. Determining metric coefficients

As mentioned earlier, it is necessary to identify several vehicles to obtain the metric coefficients in the first few frames. For this purpose, the method presented in [35] is used which functions based on the BoW and SVM. The advantage of this method is that it can work well with a small amount of training data. Figure 5 shows the result of this algorithm for some images.

Since each vehicle is available in several frames and it may be misclassified in some frames, we save the classification results for each vehicle. Finally, for each vehicle, we combine the decisions made from different frames based on majority voting. In other words, the class with maximum votes is assigned to the vehicle, as formulated in Eq. (5):

$$C = \operatorname{argmax}(\sum_i \text{class}_i). \quad (5)$$

In this equation, i represents the number of frames where the vehicle is tracked and class_i represents the winner class in the i th frame.

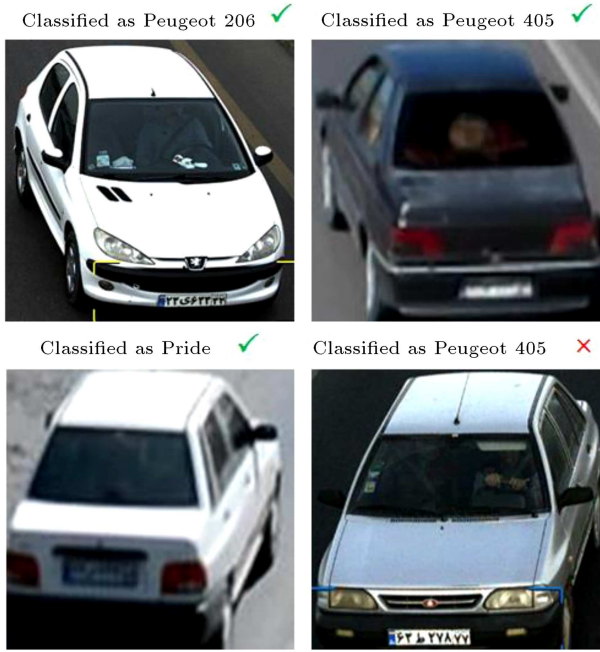


Figure 5. BoW classification results.

To calculate the metric coefficients, we only need to identify two or three types of vehicles. By using the proposed method, K samples are identified for two popular passenger cars in Iran, Pride 131 and Peugeot 405. Then, we obtain the metric coefficients by dividing the virtual dimensions by the actual dimensions of them. Since these two vehicles have high traffic on roads, the identification of K samples ($K \geq 5$) does not take much time. After identifying candidates for each vehicle class using the BoW method, we construct a matrix for each class according to Eq. (6):

$$pride = \begin{bmatrix} W_{pride\ 1} & \cdots & W_{pride\ K} \\ H_{pride\ 1} & \cdots & H_{pride\ K} \\ L_{pride\ 1} & \cdots & L_{pride\ K} \end{bmatrix}, \quad (6)$$

$$p405 = \begin{bmatrix} W_{p405\ 1} & \cdots & W_{p405\ K} \\ H_{p405\ 1} & \cdots & H_{p405\ K} \\ L_{p405\ 1} & \cdots & L_{p405\ K} \end{bmatrix},$$

where W_{priden} , H_{priden} , and L_{priden} are the estimated width, height, and length for the n -th car of the class pride. The same measures are obtained for the other class.

After completing matrices, the lengths, widths, and heights in each matrix that do not match the other samples are considered as outliers and are deleted. The dimensions of the class on the motion plane are determined by averaging the remaining values in each row. In the experiments, it was found that the BoW method had a maximum error of 20% in determining the type of vehicle. The reason for removing outliers from measurements is to prevent the impact of wrong candidates on dimension estimation.

After obtaining the dimensions of the vehicles on the motion plane, the metric coefficients are determined as follows:

$$\begin{aligned} M_{L-pride} &= \frac{L'_{pride}}{L_{pride}}, M_{W-pride} = \frac{W'_{pride}}{W_{pride}}, \\ M_{H-pride} &= \frac{H'_{pride}}{H_{pride}}, M_{L-p405} = \frac{L'_{p405}}{L_{p405}}, \\ M_{W-p405} &= \frac{W'_{p405}}{W_{p405}}, M_{H-p405} = \frac{H'_{p405}}{H_{p405}}, \\ M_L &= \frac{M_{L-pride} + M_{L-p405}}{2}, \\ M_H &= \frac{M_{H-pride} + M_{H-p405}}{2}, \\ M_W &= \frac{M_{W-pride} + M_{W-p405}}{2}. \end{aligned} \quad (7)$$

In Eq. (7), L', W', H' are the actual dimensions and L, W , and H are the virtual values obtained by averaging the values of Eq. (6) after removing the outliers. M_L, M_H, M_W are the metric coefficients of length, height, and width. Upon multiplying these coefficients by the dimensions of each vehicle on the motion plane, the actual metric dimensions of the vehicle are calculated.

Peugeot 405 and Pride cars that are identified by the BoW method are two popular vehicles in Iran, which are different in size. This difference in the dimensions and prevalence of these cars on roads helps the algorithm produce better results.

3.3. Phase III: speed and dimensions estimation

The initial parameters required for the algorithm, which include vanishing points and metric coefficients, are calculated at the first and second phases. At the third phase, these parameters are used to calculate the dimensions and speed of each vehicle. Figure 6 shows the flowchart of this phase.

To estimate the speed, we need to get the car displacement between the two consecutive frames (Δd). Due to the high frame rate of roadside cameras and the fact that the speed of cars on the road does not change all of a sudden, Δd between successive frames is almost the same. Thus, to increase the accuracy of estimating the speed and dimensions of the car, we track each vehicle and create histograms for the dimensions and Δd of the vehicle. Each histogram is analyzed and the index in which the histogram is maximum is selected as $L/W/H$ or Δd accordingly. These values are in pixel. To determine the final measurements in meter, these are multiplied by the metric coefficient as in Eq. (8):

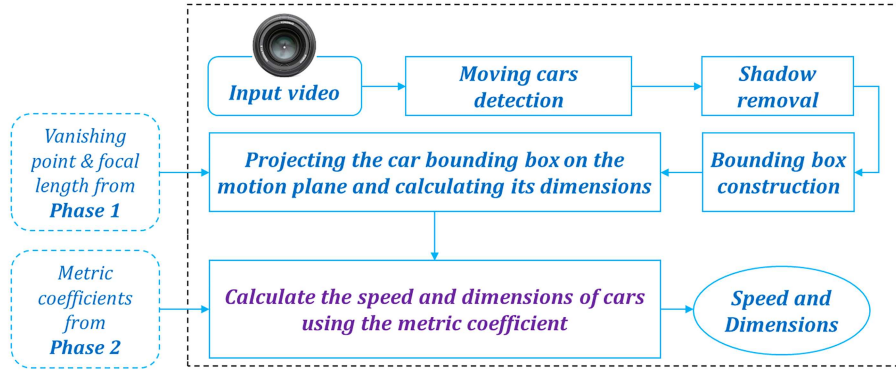


Figure 6. Flowchart of the third phase for speed and dimension estimation.

$$L' = M_L \times \operatorname{argmax}(Hist_L),$$

$$H' = M_H \times \operatorname{argmax}(Hist_H), \quad (8)$$

$$W' = M_W \times \operatorname{argmax}(Hist_W),$$

$$\Delta d' = M_L \times \operatorname{argmax}(Hist_{\Delta d}).$$

In this equation, M_H , M_W , M_L are metric coefficients, $Hist_H$, $Hist_W$, $Hist_L$, $Hist_{\Delta d}$ are dimensions and displacement histograms, L' , H' , W' , and $\Delta d'$ are the final metric parameters for each vehicle. Finally, Eq. (9) is used to estimate the speed in km/h:

$$V = \frac{3.6 \times \Delta d'}{\Delta t}, \Delta t = \frac{1}{FR}, \quad (9)$$

where FR represents the Frame Rate and Δt is the time interval between the two consecutive frames.

4. Experimental results

The dataset used to evaluate the proposed algorithm consists of two parts, a set of videos and a set of images. We captured the videos of several roads in Iran with an angle similar to that of roadside cameras. To estimate the actual speed of vehicles in these videos, we used a speed camera from Iranian Police (Figure 7). The average speed of vehicles varies depending on each road. The first row of the images in Figure 7 illustrates a sample of laser camera images and the speed recorded by it. The second row shows the traffic graph of some vehicles in terms of speed from the videos recorded in Shahrood city.

Figure 8 shows the experimental setup for image acquisition and the results of the 3D box construction. The first row shows the RGB and laser camera setup for video acquisition and speed recording. The second

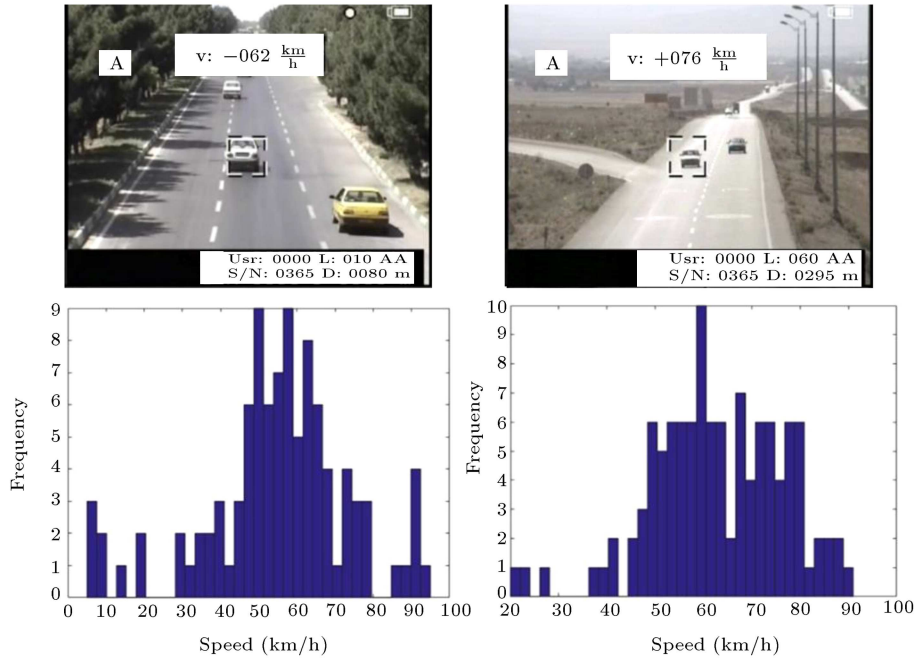


Figure 7. Top images showing the speeds recorded for vehicles by a laser camera. Bottom images showing the histograms of vehicle speeds on two roads.



Figure 8. Top row: RGB and laser camera setup for video acquisition and speed measurement. Bottom row: 3D bounding box fitted by the proposed method for two sample frames.

row shows the formation of the 3D bounding box for two frames using the proposed method. Manual data acquisition is conducted on roads of the two cities of Iran, Shahrood and Bastam.

For the identification of cars, we collected 5000 images, most of which were captured from the road and partly provided by the police. Some images are shown in Figure 5. Vehicles belong to 12 distinct classes. Table 1 shows the results of identifying these vehicles using the BoW method. According to the table, despite the camera's distances from vehicles, this method performs accurately. As mentioned in Section 3.2.4, we only use the two passenger cars, Pride 131 and Peugeot 405, to calculate the metric coefficients. These two vehicles have different dimensions and are more popular in Iran. Although other classes can also be used, these two classes are better choices considering the process of calculating metric coefficients.

This study evaluated our method based on the provided video dataset and achieved the average er-

ror of 1.15 km/h in speed estimation and 2.3% in dimension estimation (length of vehicles). Table 2 shows the processing time required for each step of the proposed method on video frames. These times are obtained by a personal computer featuring a Core i7 CPU at 2.6 GHz and 16 GB of RAM. The code is implemented in MATLAB and, therefore, these times are somewhat long. Processing time is linearly related to the amount of traffic as each new vehicle requires separate processing. Therefore, we provided the required times in two different conditions, low traffic and high traffic.

Table 3 shows the comparison of the results of our method with the related works. However, it must be stated that the datasets are different. Among the compared algorithms, methods [18,19,21] require the user to determine parameters such as elevation, angle of the camera, etc. A slight error in measuring these parameters causes a significant error in the camera calibration. In [14,17,20], the calibration was based

Table 1. Results of car classification by BoW.

Vehicle type	Recognition rate (%)	Vehicle type	Recognition rate (%)
Pride 131	90	L90	93
Peugeot Pars	93	Samand	89.5
Peugeot 206	99.5	Pride 141	91
Peugeot 405	83.5	Bus-Truck	85
Pride 132	85.5	Pickup	89
Peykan	95	others	87

Table 2. Required time for each stage of the algorithm.

Stage	Processing time (low traffic)	Processing time (high traffic)
Car detection	60 ms	70 ms
Shadow removal	50 ms	90 ms
3D Box construction	100 ms	300 ms
Tracking	90 ms	200 ms
Total	300 ms	660 ms

Table 3. Comparison of the proposed method and related works.

Method	Ground truth generation	Avg. error in speed estimation	Dimension estimated?
Filipiak [17]	Induction loops	2.3 km/h	No
Luvizon [18]	Induction loops	1.63 km/h	No
Sina [19]	GPS	3.3 km/h	No
Eslami [14]	Radar and Laser	1.22 km/h	No
Aghayan [21]	GPS	1.39 km/h	No
Famouri [20]	Laser	1.43 km/h	No
Asgarian [23]	Laser	–	Yes
Suggested	Laser	1.15 km/h	Yes

on the license plate size. Thus, any error in the license plate localization increased the overall error.

Our previous work [23] can only be used to estimate the dimensions. In Table 3, the results of this method are reported for the videos presented in this paper. Due to the sensitivity of this method to the shadow, the box formed for each vehicle is usually larger than its actual size which this causes the algorithm to perform poorly in terms of accuracy.

According to Table 3, the proposed method has fewer errors in speed estimation than the other methods. The proposed method error in dimension estimation is 2.3%, which is also less than the error observed for the other methods. Of course, if the test videos are taken from inappropriate angles, the error of the proposed method may be increased, especially in dimension estimation.

5. Conclusion

The application of machine vision and image processing methods to video surveillance cameras in order to estimate the speed of the vehicles can obviate the need for expensive hardware such as radar and laser on the road in the future. Moreover, video road analysis can provide more comprehensive information about vehicles. In this paper, a fully automated method for estimating the speed and dimensions of vehicles was presented. This method is based on camera calibration, vehicle recognition by BoW, and some statistical processing. To test the efficiency of the method, a video dataset and a car image dataset were prepared. To increase the efficiency of this method, it can be implemented in GPU with CUDA platform to be used in real-time applications.

References

- Mosayebi, A. and Khosravi, H. “A simple approach for real time speed estimation of on road vehicles using video sequences”, *First International Conference in applied research on Electrical, Mechanical and Mechatronics Engineering* (2016).
- Hua, S., Kapoor, M., and Anastasiu, D.C. “Vehicle tracking and speed estimation from traffic videos”, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 153–160 (2018).
- Ratajczak, R., Domański, M., and Wegner, K. “Vehicle size estimation from stereoscopic video”, *2012 19th International Conference on Systems, Signals and Image Processing (IWSSIP)*, IEEE., pp. 405–408 (2012).
- Biglari, M., Soleimani, A., and Hassanpour, H. “Part-based recognition of vehicle make and model”, *IET Image Processing*, **11**(7), pp. 483–491 (2017).
- Asgarian Dehkordi, R. and Khosravi, H. “Vehicle Type Recognition based on Dimension Estimation and Bag of Word Classification”, *Journal of AI and Data Mining*, **8**(3), pp. 427–438 (2020).
- Kim, Y. and Kum, D. “Deep learning based vehicle position and orientation estimation via inverse perspective mapping image”, *2019 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, pp. 317–323 (2019).
- Dubská, M., et al. “Fully automatic roadside camera calibration for traffic surveillance”, *IEEE Transactions on Intelligent Transportation Systems*, **16**(3), pp. 1162–1171 (2014).
- Dawson, D.N. and Birchfield, S.T. “An energy minimization approach to automatic traffic camera calibration”, *IEEE Transactions on Intelligent Transportation Systems*, **14**(3), pp. 1095–1108 (2013).
- Sochor, J., Juránek, R., and Herout, A., et al. “Traffic surveillance camera calibration by 3d model bounding box alignment for accurate vehicle speed measurement”, *Computer Vision*, **161**, pp. 87–98 (2017).
- Zhang, J., Xiao, W., Coifman, B., et al. “Vehicle tracking and speed estimation from roadside lidar”, *IEEE Journal of Selected Topics in Applied Earth Observations*, **13**, pp. 5597–5608 (2020).

11. Dong, H., Wen, M., and Yang, Z. "Vehicle speed estimation based on 3D ConvNets and non-local blocks", *Future Internet*, **11**(6), p. 123 (2019).
12. Wang, W., Yang, S., Li, Y., et al. "A rough vehicle distance measurement method using monocular vision and license plate", *2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, IEEE, pp. 426–430 (2015).
13. Xu, H. and Wang, X. "Camera calibration based on perspective geometry and its application in LDWS", *Physics Procedia*, **33**, pp. 1626–1633 (2012).
14. Eslami, H., Raie, A., and Faez, K. "Precise vehicle speed measurement for law enforcement applications based on calibrated camera with parallel standard patterns", *IET Computer Vision*, **10**(5), pp. 398–406 (2016).
15. Zhan, X., Zheng, Y., Yi, X., et al. "Citywide traffic volume estimation using trajectory data", *IEEE Transactions on Knowledge*, **29**(2), pp. 272–285 (2016).
16. You, X. and Zheng, Y. "An accurate and practical calibration method for roadside camera using two vanishing points", *Neurocomputing*, **204**, pp. 222–230 (2016).
17. Filipiak, P., Golenko, B., and Dolega, C. "NSGA-II based auto-calibration of automatic number plate recognition camera for vehicle speed measurement", *European Conference on the Applications of Evolutionary Computation*, Springer, pp. 803–818 (2016).
18. Luvizon, D.C., Nassu, B.T., and Minetto, R. "Vehicle speed estimation by license plate detection and tracking", *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, pp. 6563–6567 (2014).
19. Sina, I., Wibisono, A., Nurhadiyatna, A., et al. "Vehicle counting and speed measurement using headlight detection", *2013 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, IEEE, pp. 149–154 (2013).
20. Famouri, M., Azimifar, Z., and Wong, A. "A novel motion plane-based approach to vehicle speed estimation", *IEEE Transactions on Intelligent Transportation Systems*, **20**(4), pp. 1237–1246 (2018).
21. Khosravi, H. and Aghayan, S.A. "Real-time vehicle speed estimation from monocular camera using center of gravity tracking and implementation on the XU4 development board", *10th Iranian Conf. Machine Vision and Image Processing (in Farsi)*, Isfahan, Iran (2017).
22. Dubská, M., Herout, A., and Sochor, J. "Automatic camera calibration for traffic understanding", *British Machine Vision Conference (BMVC)*, p. 8 (2014).
23. Khosravi, H. and Asgarian Dehkordi, R. "Automatic detection of vehicle dimensions using video received from roadside cameras", *10th Iranian Conf. Machine Vision and Image Processing (in Farsi)*, Isfahan, Iran (2017).
24. Lin, C.T., Yang, C.T., Shou, Y.W., et al. "An efficient and robust moving shadow removal algorithm and its applications in ITS", *EURASIP Journal on Advances in Signal Processing* **2010**, pp. 1–19 (2010).
25. Wang, J., Zheng, H., Huang, Y., et al. "Vehicle type recognition in surveillance images from labeled web-nature data using deep transfer learning", *IEEE Transactions on Intelligent Transportation Systems*, **19**(9), pp. 2913–2922 (2017).
26. Krizhevsky, A., Sutskever, I., and Hinton, G.E. "Imagenet classification with deep convolutional neural networks", *Advances in Neural Information Processing Systems*, pp. 1097–1105 (2012).
27. Ren, S., He, K., Girshick, R., et al. "Faster r-cnn: Towards real-time object detection with region proposal networks", *Advances in Neural Information Processing Systems*, pp. 91–99 (2015).
28. Yang, L., Luo, P., Change Loy, C., et al. "A large-scale car dataset for fine-grained categorization and verification", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3973–3981 (2015).
29. Yu, S., Wu, Y., Li, W., et al. "A model for fine-grained vehicle classification based on deep learning", *Neurocomputing*, **257**, pp. 97–103 (2017).
30. Gholamalinezhad, H. and Khosravi, H. "IRVD: A large-scale dataset for classification of Iranian vehicles in urban streets", *Journal of AI and Data Mining*, **9**(1), pp. 1–9 (2020).
31. Hsieh, J.W., Chen, L.C., and Chen, D.Y. "Symmetrical SURF and its applications to vehicle detection and vehicle make and model recognition", *IEEE Transactions on Intelligent Transportation Systems*, **15**(1), pp. 6–20 (2014).
32. Daniilidis, K., *How to Compute Intrinsic from Vanishing Points*, cited 2020. Available from: <https://www.coursera.org/lecture/robotics-perception/how-to-compute-intrinsic-from-vanishing-points-jnaLs> (2018).
33. Zivkovic, Z. "Improved adaptive Gaussian mixture model for background subtraction", *Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004*, IEEE, pp. 28–31 (2004).
34. Kamijo, S., Matsushita, Y., Ikeuchi, K., et al. "Traffic monitoring and accident detection at intersections", *IEEE Transactions on Intelligent Transportation Systems*, **1**(2), pp. 108–118 (2000).
35. Eraqi, H., *Bag of Visual Words for Image Classification*, cited 2020. Available from: <http://heraqi.blogspot.com/2017/03/BoW.html> (2017).

Biographies

Hossein Khosravi studied Electronic Engineering at Sharif University of Technology in 2003. He received his MSc and PhD degrees in Image Processing

from Tarbiat Modares University in 2005 and 2009, respectively. Since 2009, he has been teaching at the Faculty of Electrical Engineering at Shahrood University of Technology. He is also the CEO of Shahaab Co. (<http://shahaab-co.com>). His research interests include neural networks, deep learning, and image processing.

Rasoul Asgarian Dehkordi graduated with a Master's degree in Electronics Engineering in 2014 from the University of Isfahan and a PhD in 2019 from Shahrood University of Technology. His research interests are machine vision and image processing, deep neural networks, pattern recognition, evolutionary algorithms, and parallel processing. During his studies and work,

he has collaborated with research groups and has done several projects in his field.

Alireza Ahmadyfard received his BSc degree in Electronic Engineering in 1989 from Esfahan University of Technology and MSc degree in Communication Engineering in 1994 from Amirkabir University of Technology. He earned his PhD in Image Processing and Computer Vision from CVSSP (Center for Vision Speech and Signal Processing) at University of Surrey in 2002. He is the Director of Electrical Engineering Department at Shahrood University of Technology. His research interest is digital signal processing, object recognition, image-based inspection, and human identification using biometrics.