



# Image projection method for vehicle speed estimation model in video system

Mallikarjun Anandhalli<sup>1</sup> · Pavana Baligar<sup>2</sup> · Santosh S. Saraf<sup>1</sup> · Pooja Deep sir<sup>1</sup>

Received: 2 March 2021 / Revised: 7 September 2021 / Accepted: 6 October 2021 / Published online: 23 November 2021  
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2021

## Abstract

Recent developments in Information and Communication Technology have facilitated a new concept of Smart City to the world. The Smart City concept is driven by technology intervention in every aspect of city life, including the most dynamic and unpredictable transport management. Intelligent transport management system (ITMS) is the most essential component of a Smart City ecosystem. The primary function is to ensure smooth and accident-free transport on the city roads. ITMS can prompt drivers of possible traffic jams; ITMS can be used to detect speed violations of vehicles. One of the primary inputs to ITMS is fed from closed circuit television (CCTV) installations on the roads. The main objective of the paper is to detect vehicles violating traffic rules especially over-speeding. The detection of over-speeding of a vehicle involves detection of vehicle, calculation and calibration of the distance traveled by the vehicle both on an image plane and real world. To calibrate the distance traveled by the vehicle, the geometric plane of the real world is projected onto the image plane. The projection onto the image plane helps in determining the actual distance traveled by the vehicle in the real world. After calibration of the distance traveled by the vehicle, speed calculation is performed. The accuracy of the algorithm to speed detection is 90.8%.

**Keywords** Vehicle detection · Calibration · Plane projection · Speed estimation · Tracking

## 1 Introduction

The development of ITS has provided many technologies for transport management, vehicle control, collision avoidance systems, emergency management services, etc. According to the reports, each year around 1.3 million accidents happen all over the world and most of them are caused due to the over speeding of vehicles. Vehicle speed detection plays a very important role in enforcing speed limits which decrease accidents and traffic jams. If the on-road vehicles keep increasing at the current rate, it is really difficult to provide required manual labor at all busy areas to control the traffic [1,2]. Monitoring the speed of the moving vehicles manually is not

feasible everywhere. Automation of speed detection is need of the hour [3,4]. Using a traffic surveillance video, we can detect the moving vehicle's speed. Traditional speed detection was using radar technology which works on the doppler effect[5–8]. The errors caused by radar technology such as cosine error, shadowing and frequency interference limit the accuracy of the speed detection. Hence, using this traditional technology is not enough in this current scenario [9]. With the rapid development in the technology, traffic management system demands computerized technology to control the problem of over speeding [10,11].

A LIDAR device keeps track of the time taken by a light pulse to hit the vehicle from the LIDAR gun and come back. Using this data, LIDAR estimates the distance between the gun and the vehicle. By making various calculations and comparing them, the speed of a vehicle can be determined accurately.

A passive camera system is usually less expensive than the active devices used by the above-mentioned methods. Imaging hardware available is of less cost and they are of high performance due to which in recent years, video-based approaches have been recommended for both vehicle speed detection and vehicle tracking [12–14].

✉ Mallikarjun Anandhalli  
malliarjun71@gmail.com

<sup>1</sup> Department of Electronics and Communication Engineering, KLS Gogte Institute of Technology, Belagavi, India

<sup>2</sup> Department of Computer Science and Engineering, Smt. Kamala & Sri Venkappa M. Agadi College of Engineering & Technology, Lakshmeshwar, Gadag, India

## Objectives

- The proposed solution has to be highly reliable on vehicle detection.
- Once the vehicle is detected, distance of the vehicle is estimated by projection method in the image plane.

## 2 Literature survey

Wu et al. [15] describe a new method for real-time automatic vehicle speed monitoring using a video camera that is based on digital image processing. It begins by presenting a simplified method for accurately mapping the coordinates in the image domain into the real-world domain, which is based on geometric optics.

The second section focuses on detecting vehicles in digital image frames in a video stream. The solution requires a single video camera and an onboard processing computer to operate and simultaneously detect vehicle speeds in multiple lanes with high accuracy. The algorithm requires that the camera is set up directly above the target road section (at least 5 meters above the road to assure satisfactory accuracy) with its optical axis tilting a certain angle downward from the highway forward direction. The calibration is done directly on the video frames based on the position of an easy-to-get vanishing point and a vehicle's known length and width and its information (upper edge position and lower edge position) in a sample image. The results are reported with a < 4% error.

Feature extraction is an important factor in detection of vehicles. Several works are referred for the feature extraction process and have been discussed in this section. Authors, Zhang et al. [16], proposed a method for feature selection that is superior than the spares-based method using  $l_2 - l_1$  regularization and authors [17] extended work of feature selection, by selecting features using reconstructed data and other pairwise constraints. The features are selected in unsupervised manner to minimize selection error, to preserve graph and side information's. Authors considered aforementioned parameters in feature selection and obtained robust solutions using robust loss function that interpolates between  $l_1$  and  $l_2$  norms. Further, feature selection is also made by using regression model and generalized uncorrelated constraints [18] by adopting the  $\sigma$  form regularization for interpolating between F-form and  $l_2 - l_1$  norms.

Vehicle length detection is an important parameter in determining speed. Various methods are reported in the literature. Ki et al. [19] proposed a system to measure speed using a double loop detector. The authors proposed a use of neural networks with a back-propagation algorithm to improve the accuracy of inductive loop detectors. The neural network classifies five types of vehicles based on inputs of variation

rate of frequency and occupancy time. The algorithm has an efficacy of 91.7%.

Jing-zhong et al. [20] proposed a system to find the speed of the vehicle by comparing video frames. The proposed technique uses background differences in images. Pixel-based comparison is done on the selected background image with the image of a moving vehicle. The background difference will be calibrated for the speed of the vehicle. This method is influenced by the quality of images.

Rahmin et al. [21] proposed to detect the speed of a vehicle using a camera feed. The algorithm involves an extraction of the object(vehicle) using the frame difference technique. The object detected is mapped onto a 3D space and vehicle speed is estimated with an error of  $\leq 1.7$  km/h. The assumptions made are that all the camera parameters are known, vehicles move on flat ground, and they are moving forward. Estimation of the accuracy of vehicle velocity is constrained by the timing resolution and displacement resolution in pixels on the images.

Pelegri et al. [22] proposed a system that uses GMR magnetic field grade sensors placed on the pavements on the road. The signals of vehicular movement are captured and the speed and the length of the vehicle can be captured in real time. Discrete cross-correlation is used to detect the time interval  $t$  the time difference in the signal between the front wheel and back wheel crossing the sensors.

Pornpanomchai et al. [23] proposed the vehicle speed estimation system by extracting frames from a video with a reference point and an ending point. The authors proposed a system to extract, enhance and detect vehicles in video frames by using the frame difference technique. The system reports the usability and effectiveness of the technique with 70 s as the time to report the speed of the vehicle. The authors conclude with the usability of the system based on the length of the video to be processed, the length of the vehicle and the stability of the camera to record good and noise-free video.

Pelegri et al. proposed a system that uses GMR which is a magnetic sensor to detect the object which is the vehicle in this paper and to monitor the speed of that detected vehicle. The magnetic perturbation is detected which is caused by the vehicle by using two GMRs which are located on the highway. The signals from the sensor are generated when a vehicle moves above the board of the circuit, and then, they are sent to a microcontroller to process and then the speed and length of the vehicle are obtained. This can replace the ultrasonic sensors which are expensive.

Pornpanomchai et al. [23] proposed the vehicle speed estimation system by applying the image processing technique. This research contemplates to develop detection of vehicle speed. This system used IBM/PC which is connected to the uncalibrated camera. The five components of the system are image enhancement, acquisition, region of interest segmentation, analysis and speed estimation.

Authors, Kumar et al. [24], proposed a method to detect, track and estimate the speed of the vehicle. Detection and tracking of the vehicle use the parameters such as height, position and width of the vehicle instead of extracting features of a vehicle. The parameters used as a feature extractor require less computation and memory. The aforementioned parameters are stored in the database to estimate the speed of a vehicle.

Authors, Javadi et al. [25], proposed a mathematical model using the movement pattern vector for estimating the speed of a vehicle. The speed of the vehicle is estimated by using four intrusion lines and the frame was captured simultaneously by a camera with 50 fps and a smartphone device with fps of 30. The probability density function is used to improve the speed of the vehicle with an average error rate of 1.77% in 50fps and 2.17 in 30fps.

The authors, Lu et al. [26], proposed a method to estimate the speed of the vehicle by considering video sequence and without extracting the features. The frame difference method is applied to a region of interest, resulting in projection histograms and the selection of a group of key bins to represent vehicle motion.

## 2.1 Problem statement

The LIDAR- and RADAR-based speed estimation requires the hardware equipment which is not cost-effective and have the following limitation:

1. Cosine error as mentioned in the manuscript.
2. A false reading is produced when the pulse reflects off for example a wing mirror, hits a stationary reflective object and then returns to reflect off the mirror a second time.
3. Radar speed guns do not differentiate between targets in traffic, and proper operator training is essential for accurate speed enforcement.

Hence, the need to replace them with the systems that are automated i.e., they give more accurate outputs, less expensive and exclude human intervention when not needed.

“A real-time estimation of vehicle’s speed based on geometric projection method and image processing for better analysis and understanding of traffic scenes.”

Nowadays, the road accidents are increasing every day. The rate of fatality is also increasing. One of the major concerned is road accidents which affects many factor, if it is not under the control. Therefore, the system for the detection of the speed is introduced to reduce accidents and to increase road safety. To replace the traditional speed detection method which is based on RADAR equipment and speed guns are not preferred due to the cosine errors. These speed guns work only when the direction of the gun is on the path

which is direct of traffic. This error is significant and this will affect the accuracy of the system.

## 3 Proposed method

*Background* Vehicle detection is said to be the main and important part of the Intelligent transportation system (ITS) [20]. Many other approaches which are included for the detection are radar, video image detection system, infrared detection, ultrasonic detection, induction line detection, acoustic array detection and so on. There are various new vehicle detection techniques from videos that are captured are discovered [27,28]. Using this vehicle detection, many other information about the vehicles can be obtained [20,29].

After studying or going through many of the research papers and journals, it has been observed that existing systems the technique of detecting vehicle using a static camera which is placed at some height above the ground [30]. Similarly, some other research papers have proposed finding the speed by first drawing two lines that are virtual and then calculating the time traveled by each vehicle between these virtual lines. The proposed system has a low cost and the accurate speed of the vehicle is obtained using computer vision than any other system. In this system, some of the standard datasets which are readily available have been used [31].

For vehicle detection, vehicle motion analysis is much needed. Here, when it comes to the detection of the vehicle, it is focused on motion detection. Moving target detection is done by a motion detection algorithm. The main algorithm for motion detection is background subtraction, frame difference and optical flow method. This paper uses a background detection algorithm to detect the motion patterns in the video. This method includes: motion detection, vehicle tracking and vehicle speed calculation.

Firstly, the video is captured using a static camera which is placed at some height above the ground for the data input. Then, the frames are extracted for further processing and to detection vehicles. The flowchart of the system design is shown in Fig. 1.

The detailed architecture of the proposed method is as shown in Fig. 2 wherein the input video is taken first, then this video is extracted into frames. Each frame is treated as the image which undergoes pre-processing. To minimize some of the errors such as bold shadow which detects the shadow as a solid object, pre-processing is used. Pre-processing is performed to enhance the quality of the image at a lowest level of abstraction which in turn enhances some of the features of the images extracted for further processing.

The main objectives involved in this proposed system design and implementation are

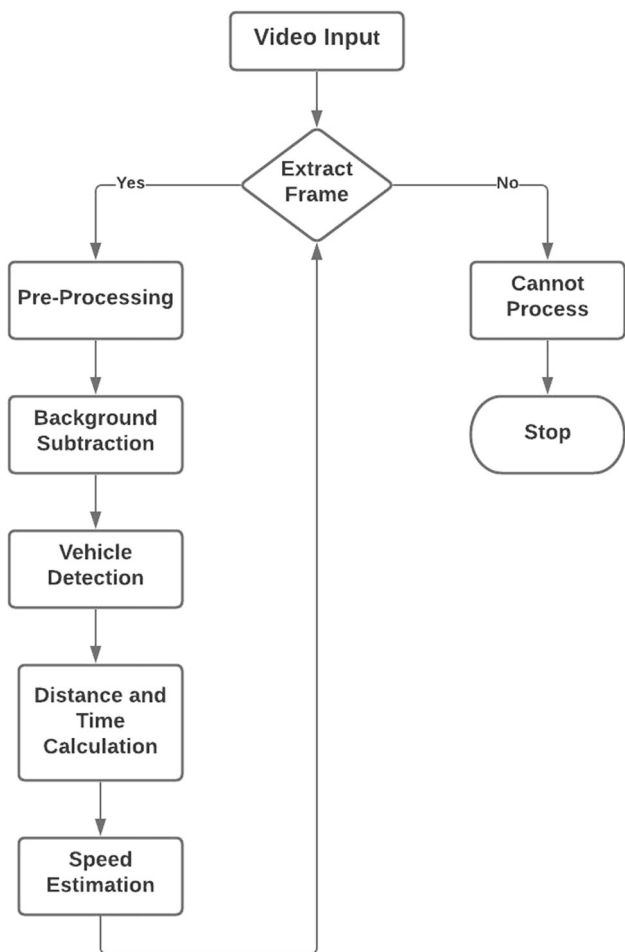


Fig. 1 Flowchart of the proposed method

- Vehicle detection.
- Distance calculation.
- Speed calculation

### 3.1 Vehicle detection

Vehicle detection plays a major task in the estimation of the speed of a vehicle. The accuracy of detection helps in achieving the estimation of speed with less error rate. Many methods exist in detecting the vehicle in the image processing domain, but when considering the parameters like complexity, readability and accuracy, the background subtraction methods can achieve better results compared with other traditional methods.

Several methods already exist with good accuracy like Gaussian distribution, adaptive median, morphological background estimation and nonparametric background modeling paradigm. Considering the video in a stationary camera [32,33], the motion detection algorithm performs well in sep-

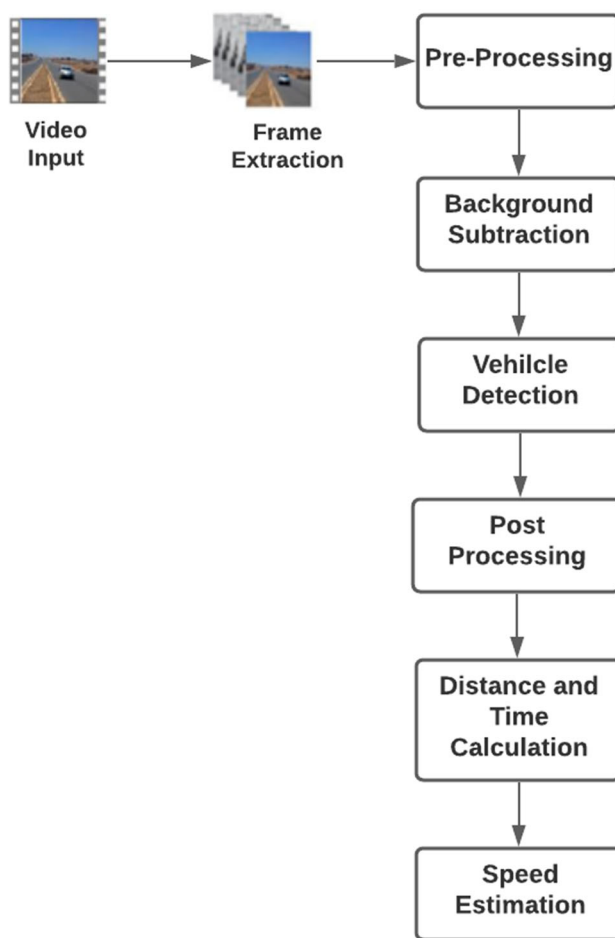


Fig. 2 Proposed Architecture for the vehicle speed estimation

arating the foreground image from the background based on the changes that are taking place in the foreground.

After the detailed study on the background subtraction, the Gaussian mixture model is considered in the proposed method. In background subtraction, the background models are initialized using the first “K” frames at a pixel level. Where K denotes the number of the Gaussian models in a GMM and  $X_i$  denotes a particular pixel located on  $(x, y)$  in the  $i$ -th frame I as in (1).

$$X_1, \dots, X_K = I(x, y, i) : 1 \leq i \leq K \tag{1}$$

Each pixel is modeled using K Gaussian mixture. Then, the probability function of the given pixel at time t is given by:

$$P(X_t) = \sum_{i=1}^k \omega_i \cdot \eta(X_t, \mu_i \sigma_i^2)$$

where  $X_t$  is denoted as pixel observed at time t,  $\omega_i$  is the weight related to the  $i$ -th Gaussian distribution with mean

$\mu_i, t$  and standard deviation  $\sigma_i^2$ .  $\eta$  is a Gaussian probability density function given as

$$\eta(X_t, \mu_i \sigma_i^2) = \frac{1}{(2\pi \sigma_i^2)^{1/2}} e^{-\frac{X_t - \mu_i^2}{2\sigma_i^2}}$$

in which  $\mu_i$  is the mean of pixel value in the  $i$ -th image sequence and  $\sigma_i$  is the user parameter with a value set between 10 and 20 randomly. Once the models are initialized, the foreground and background models are separated using:

$$B = \operatorname{argmin}_b \left( \sum_{k=1}^b \omega_k > T \right)$$

where the first  $B$  distributions are considered as the background models if the sum from  $\omega_1$  to  $\omega_b$  exceeds a threshold ( $T$ ), and the distributions which are left out are considered to be foreground models.

Let the color model and the depth model be  $P(ct)$  and  $P(dt)$  as follows:  $P(X_t) = P(c_t)P(d_t)$

Now the background/foreground segmentation with the probabilistic background models is done. First, the color model  $B_c$  and depth model  $B_d$  classify the observed pixels  $c_t$  and  $d_t$  at time  $t$ , respectively. A pixel finds a matched Gaussian model using the Euclidean distance:  $\|x_t - \mu_i\| \geq k$

A pixel is classified as one of the following 3 category if the match is found in respective model.

Category 1: pixel is classified as background, if matched model is the background model. Category 2: pixel is classified as foreground, if matched model is the foreground model. Category 3: pixel belongs to foreground, if no match.

In the case of color value  $c_t$ , once pixels are classified as background or foreground at a time “ $t$ ,” the inequality for the matched Gaussian distribution can be used to compute pixel value using:

$$\theta \cdot \eta(c_t, \mu_i \sigma_i^2) \geq \text{Maximum pixel value}$$

in which  $\theta$  is a constant to scale the Gaussian probability density Function values. Thus, the final BGS (background subtraction) results  $R_{final}$  are evaluated if:

- $R_c(x, y) * R_d(x, y) > q$  Maximum pixel value,
- then  $R_{final}(x, y)$  is foreground,
- else then  $R_{final}(x, y)$  is background.

### 3.2 Distance calculation

In the proposed system, to calculate the virtual distance traveled by the vehicle, Kalman filter is used for the tracking which draws imaginary lines on the output video of the vehicles after detection on the image plane, which is called entry

point detection line and exit point detection line (assuming any two points with respect to the centroid of the vehicle). The distance traveled by each vehicle between these two lines in the video is used to calibrate the actual speed in real-time.

These moving objects which are detected are outlined with the bounding box to identify and also determine the centroid of the detected object. Centroid point is used to determine the distance traveled by the vehicle that is detected. This distance is calculated using the Euclidean distance formula. This distance is further used to estimate the speed of the detected vehicle. The centroid of each detected vehicle is tracked using some of the suitable tracking algorithms. The centroid of each vehicle detected can be calculated using

$$(x_c, y_c) = \left( \frac{(x_1 + x_2)}{2}, \frac{(y_1 + y_2)}{2} \right)$$

where  $(x_c, y_c)$  is the center of the vehicle detected.

The entry frame number and exit frame number are noted when these detected vehicles coordinate lies within entry point detection and exit point detection range, respectively. Tracking the entry and exit frames for each moving vehicle by storing the frame number, where the centroid of the object lies in the entry point detection range and exit point detection range, respectively. The Euclidean distance using the centroids of the entry frame and exit frame from the entry point detection and exit point detection lines, respectively, can be calculated.

### 3.3 Speed calculation based on geometric projection

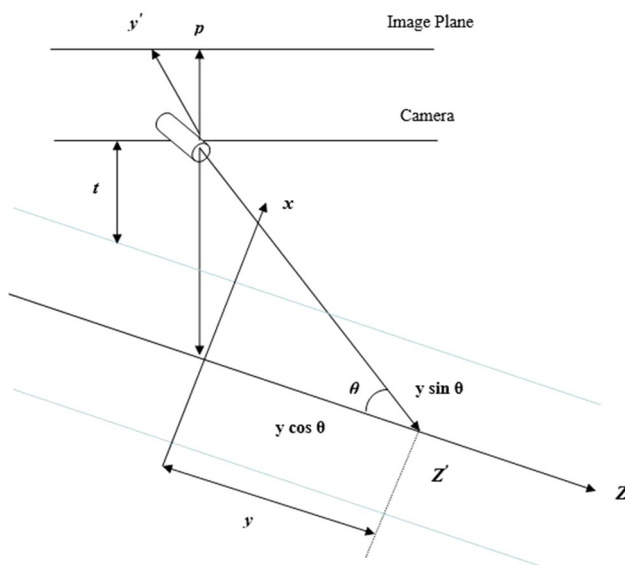
The speed of the vehicle is calculated using the virtual distance traveled by the vehicle and some other parameters are included which are shown in Fig. 3. Based on the geometric projections into the camera plane, if a vehicle is moving from point “ $p$ ” to point “ $y$ ” with a displacement of  $y$  on the image plane as shown in Fig. 3, then distance traveled by the vehicle is given by the equation below.

$$y = \frac{ty'}{f \cos \theta - (y' + p) \sin \theta} \tag{2}$$

where “ $f$ ” is the focal length of the camera which is obtained from camera calibrations and “ $t$ ” is the distance between the camera and the vehicle in the direction parallel to the optical axis. Then, the time interval  $T$  for the displacement can be calculated using the equation below.

$$T = N * F \tag{3}$$

where “ $N$ ” indicates the total number of frames, “ $F$ ” is the time required to capture each frame. If the camera exposure



**Fig. 3** Model for speed calculation

time is “ $T$ ” for the displacement and the size of the pixel in the horizontal direction is  $S_x$ , then the speed  $v$  of the moving vehicle is given by

$$v = \frac{ty'S_x}{Tf \cos \theta \left[ 1 - \frac{S_x}{f} (y' + p) \tan \theta \right]} \quad (4)$$

where  $S_x$  is called CCD pixel size which can be obtained from internal parameters of the camera and it is given by the manufacturer’s data sheet.

If  $f \gg S_x$  and the angle  $\Theta$  is less than  $45^\circ$ . Then, the speed “ $v$ ” of a vehicle can be simplified as shown below

$$v = \frac{ty'S_x}{Tf \cos \theta} \quad (5)$$

### 3.4 Moving object tracking using Kalman filter

We have used a background subtraction algorithm for the detection of vehicles. Through the background subtraction algorithm, we get the central points of the detected vehicles. Due to some of the limitations of background subtraction algorithm such as occlusion can be seen, positions of the vehicles which are obtained are wrong sometimes. To avoid erroneous results and to overcome the limitation, the Kalman filter tracking method is applied to get more accurate positions of the vehicles. Kalman filter is a framework for predicting a process’s state. Prediction and correction are the two important and main steps in Kalman filtering.

## 4 Results and discussions

### 4.1 Implementation details

As described before, the data which is taken as video is extracted into frames and then each frame is processed by pre-processing. Pre-processing is used to minimize the errors such as shadow which is wrongly determined as the object. We use contrast and brightness adjustment to do pre-processing. The algorithm is implemented using OpenCV C++ in visual studio. Results of background subtraction which is used to separate the foreground image from a background model for each frame and motion estimation to detect the vehicles are shown in Fig. 4. In Fig. 5, the background subtraction results of two different video sequences are shown, i.e., video sequence 1 and video sequence 2. For each object that is detected, a rectangular bounding box is given to identify. As discussed earlier, virtual lines are drawn on the output video of vehicles detected on the image plane to detect the distance traveled by the vehicles based on the geometric projection method and as a result speed is calculated. The speed of each vehicle is detected using the equations which are discussed earlier. The speed of each vehicle is shown at the output window. Figures 6 and 7 are the results of vehicle detection, tracking and speed estimation. Various videos were recorded with prior knowledge of speed running on the road. The speed estimation results are dependent on the results of the tracking, wherein each of the vehicles is tracked with a dedicated tracker. The distance traveled by each of the vehicles in terms of a pixel is defined by the tracker. This distance traveled helps in the estimation of the speed of the corresponding vehicle. Then, the same procedure of storing the frame number and a point. So using these two points which are noted we will be able to find the virtual distance traveled by the vehicle. Then, the frame numbers which are stored are used in the calculation of the time of displacement of the vehicle. Once the calibrated vehicle distance and time are calculated then speed estimation can be calculated.

As discussed and depicted in Fig. 4, the results of foreground pixels detection are represented of different videos. From the results of background subtraction, one can make out that the salt and pepper noise present should be minimized. The salt and noise present in the results are reduced by morphological operations which will lower the false positives.

Figures 6 and 7 are the results of frames of different video datasets. The speed of each vehicle is calculated and is shown in Figs. 6 and 7. The speed of a vehicle is calculated by considering the calibrated distance which will help in the estimation of speed with high accuracy.

## 4.2 Datasets

For experimental purposes, the algorithm was run on several video inputs to check the performance. In the paper, several video sequences represent typical situations that were not addressed in many other systems for the improvement of the accuracy of the system results. The datasets which are considered in this paper are with approximately 30fps frame rate. The tested videos were downloaded from [http://i21www.ira.uka.de/image\\_sequences/](http://i21www.ira.uka.de/image_sequences/) and <https://www.svcl.ucsd.edu/projects/traffic/>, and ten video datasets were recorded with prior knowledge of vehicle speed near a highway.

## 4.3 Performance evaluation

Few of the parameters are set to measure the quantitative analysis for performance evaluation of the proposed system, which also measures sensitivity, specificity and accuracy based on the confusion matrix. Mathematically, the quality metrics is given as:

### 4.3.1 Sensitivity

It is defined as the ratio of visible objects present in the scene to the accurately found object by the algorithm. This parameter gives the number of positive specimens accurately found. Sensitivity is directly proportional to the true positive element.

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

### 4.3.2 Specificity

It is defined as the ratio of a static object which is present in the frame to the accurately identified static object by the proposed algorithm. This parameter gives the number of negative samples that are correctly identified. Specificity is directly proportional to the true positive element.

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FN}}$$

### 4.3.3 Accuracy

The accuracy of the proposed system is denied based on the performance which includes sensitivity and specificity parameters of the system.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FN} + \text{FP}}$$

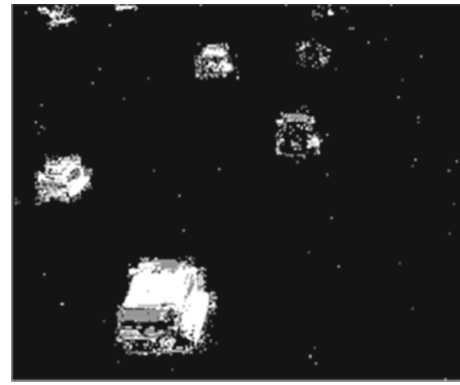


Fig. 4 Video sequence 1

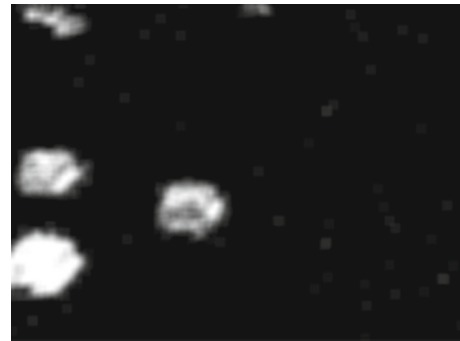


Fig. 5 Video sequence 2

## 4.4 Vehicle detection results

In normal situations, all the state-of-the-art algorithm gives good results, but in other environmental conditions like rain, fog, snow etc., it is difficult to get accurate and good results. It may fail either to detect the vehicle or track the vehicle, and hence, motion calculation of the object leads to varying results. The proposed algorithm is effective even in a complex situation such as trivial conditions. The qualitative results were obtained through the evaluation process of ground truth image and algorithm output files as input; hence, result can be accurate and average. The testing of an algorithm is done on different videos in different environmental conditions. The work is carried out in two phases, the detection of an object and finding its distance traveled and speed. The proposed algorithm is run on 3.6 GHz processor, we have used Visual Studio 2019, and the code is written in Opencv C++ and is tested on different video sequences. The proposed work can execute with 25–30FPS without any optimization. The results of the proposed work are discussed in a quantitative and qualitative way.

Table 1 depicts the results of vehicle detection for different video datasets. The number of true positives, false negatives and false positives is noted. As per the results of the proposed system with respect to the speed estimation, the results

**Fig. 6** Results of vehicle detection and speed estimation**Table 1** Results of vehicle detection

Sl. no.	Resolution	Fps	Climate	Vehicles detected (TP)	No. of vehicles present	Missed vehicles (FN)	Falsely detected vehicles (FP)
1	320 * 240	10	Sunny (low resolution)	15	15	0	0
2	320 * 240	15	Sunny (low resolution)	35	35	0	2
3	640 * 480	25	Cloudy	29	30	1	2
4	320 * 240	30	Sunny (high resolution video )	45	45	0	1
5	704 * 480	30	Rainy	24	25	2	2
6	704 * 480	30	Sunny complex (multiple )	45	45	0	1

are tabulated in Table 2 for the analysis of estimated vehicle speed, actual speed and error between them for different videos. As per the results, speed of the vehicle is estimated with a minimum error and is acceptable. Actual speed of each vehicle was noted while capturing the video and was tested against the estimated speed and same is plotted for four videos in the Figs. 8, 9, 10 and 11, respectively.

To compare and analyze the error rate among different videos and vehicles performance with the proposed method, graphical representation is made as shown in Fig. 12.

#### 4.5 Computational cost

The proposed method is implemented using Opencv c++ in visual studio as aforementioned. To evaluate the proposed method, computational cost of each operation is calculated as shown in Table 3. The operations included for computational cost are:

1. Pre-processing.
2. Background subtraction.
3. Distance calculation and speed estimation.
4. Tracking.



**Table 2** Tabular representation of estimated speed, actual speed and error

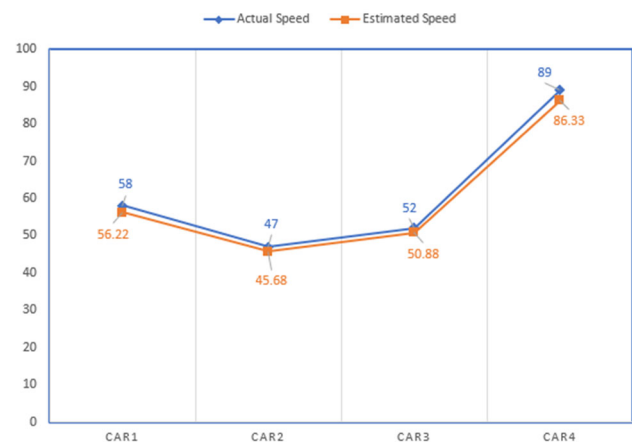
Car no.	Speed calculated in Kmph	Actual speed in Kmph	Error %	Car no.	Speed calculated in Kmph	Actual speed in Kmph	Error %
<i>Video 1</i>				<i>Video 2</i>			
1	56.2293	58	3.05	1	112.5	112	0.44
2	45.6863	47	2.79	2	107.244	115	6.7
3	50.8823	52	2.14	3	102.25	105	2.6
4	86.33	89	3	4	69.98	72	2.80
<i>Videos 3</i>				<i>Video 4</i>			
1	98.33	100	1.65	1	104.86	107	2
2	84.65	97	12.7	2	109	112	2.6
3	87.77	78	12.5	3	99.34	102	2.607
4	103.76	105	1.18	4	79	85	7.05
<i>Videos 5</i>				<i>Video 6</i>			
1	100.58	103	2.34	1	95.36	98	2.69
2	104.56	105	2.30	2	89.01	90	1.10
3	112.12	113	0.77	3	81.23	85	4.43
4	97.89	99	1.12	4	92.25	95	2.89
<i>Videos 7</i>				<i>Video 8</i>			
1	54.89	56	1.98	1	100.12	102	1.84
2	84.26	87	3.14	2	93.38	94	0.65
3	61.39	67	8.37	3	102.23	109	6.21
4	69.58	72	3.36	4	82.56	89	7.23

**Fig. 7** Results of vehicle detection and speed estimation of video sequence 5

In Table 3, the computational cost of each operations is determined and noted in ms. The overall cost to process one frame is 86.1 ms, which is comparably effective.

#### 4.6 Repeatability and reproducibility

To check the repeatability, experiments are conducted to obtain independent test results in a short period of time in the same location, with the same measurement proce-

**SPEED ESTIMATION FOR VIDEO1****Fig. 8** Video 1 speed estimation

sure, and the same measuring instrument. Table 4 shows the experimental results where the video sequence is considered in different time span on the same vehicle, same location, with same equipment for estimating speed of a vehicle. The repeatability achieved with the proposed method is 100%.

To test the reproducibility in estimating the speed of a vehicle, the proposed method is tested on standard dataset in different climatic conditions like rainy, complex background,

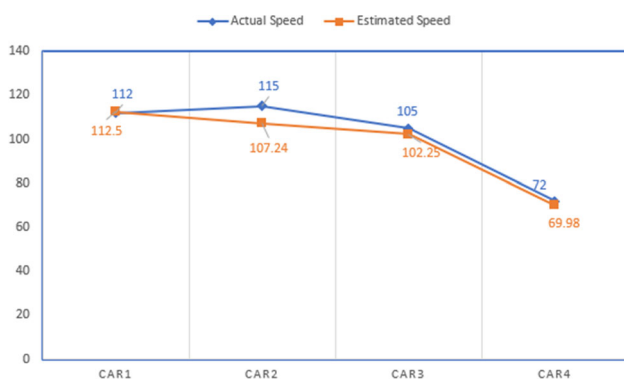
**Table 3** Computational time for real-time operations

Sl. no.	Real-time operations	Computational time (in ms)
1	Pre-processing	0.0163
2	Background subtraction	0.0319
3	Distance calculation and speed estimation	0.0204
4	Tracking	0.0175
Total time		0.0861

**Table 4** Repeatability on the proposed method

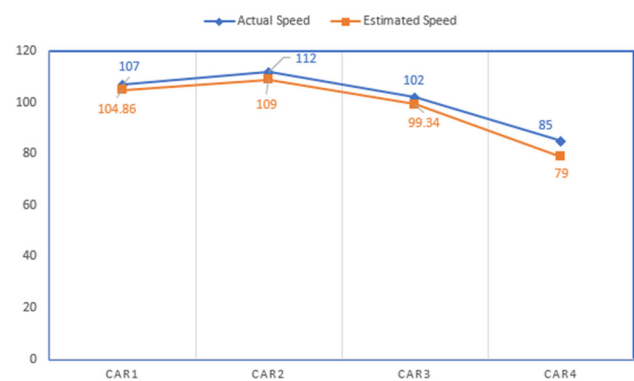
Experiment	Actual speed	Estimated speed	Error %
Video sequence 1 @ time t1	50.8823	52	2.14
Video sequence 1 @ time t2	50.8823	52	2.14

**SPEED ESTIMATION FOR VIDEO2**



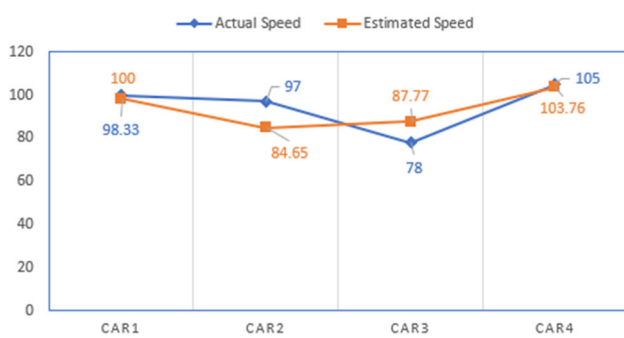
**Fig. 9** Video 2 speed estimation

**SPEED ESTIMATION FOR VIDEO4**



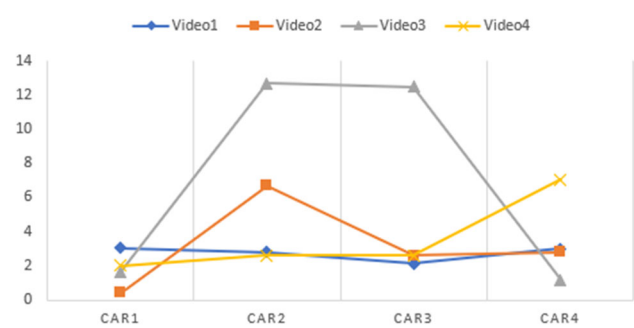
**Fig. 11** Video 4 speed estimation

**SPEED ESTIMATION FOR VIDEO3**



**Fig. 10** Video 3 speed estimation

**ERROR RATE**



**Fig. 12** Error rate of each video using the proposed method

highway, city limits and different individuals. Table 5 gives detection accuracy on different climatic conditions. Experiments conducted on these datasets give us satisfactory results in detection which is dependency factor for speed accuracy rate of a vehicle.

$$\text{Reproducibility} = \left( \frac{\text{Operation Variation}}{\text{Total Variation}} \right) \times 100 \quad (6)$$

### 4.7 Comparative analysis

For testing, the speed of the moving vehicles is known prior which acts as the ground truth for our evaluation. Table 2 represents the comparative study of the speed of the vehicle with the actual and estimated value. Different videos were considered to check the reliability and accuracy of the vehicle speed.

**Table 5** Reproducibility of the proposed method

Operation variation no.	Location type	Detection accuracy
1	Complex background	91.64
2	Highway	100
3	City limit	89.4
4	Rainy	100

**Table 6** The average errors for different methods

	Method [34]	Method [35]	Method [36]	Our method
Average error (%)	6.1	1.9	2.17	3.05

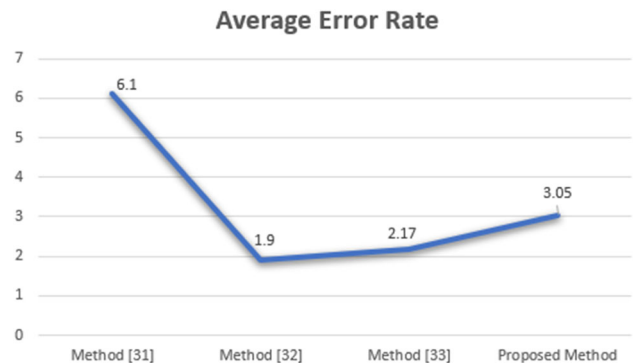
Table 6 depicts the average error of four different vehicle speed estimation algorithm.

Method [34]: authors proposed a method to estimate the velocity of the vehicle using an uncalibrated camera. Detection of vehicle is performed using the background compensation algorithm to eliminate noise in the background using horizontal and vertical histograms. The algorithm has achieved an average compensation error rate of 6.1%. The proposed method is not using the camera calibration parameters for the estimation of the speed. Method [35]: authors proposed a method to detect the speeding of vehicles using the image processing techniques over the input image sequence captured from the fixed-position video camera. Detection and tracking are obtained in the consecutive frames and license plate region is extracted from the color information. The authors achieved an error rate of 1.9%. But the authors stated that the proposed method will fail to perform when the shadow in the frame increases.

Method [36]: authors proposed a mathematical model using the movement pattern vector for estimating the speed of a vehicle. The speed of the vehicle is estimated by using four intrusion lines and the frame was captured simultaneously by camera with 50 fps and smartphone device with fps of 30. Probability density function is used to improve the speed of the vehicle with an average error rate of 1.77% in 50 fps and 2.17 in 30 fps considering the consecutive intrusion line.

It can be seen from the results that our method is almost accurate to the results obtained without prior considering the virtual lines as that of other Method [34], Method [35] and Method [36]. Method [35] and Method [36] present lower error rate only under high speed conditions and its error rate increases as the speed slows down.

To visualize the comparison of propose method, graphical representation is plotted in Fig. 13 on the error rate with different methods. From the graph, it can be seen that the

**Fig. 13** Graphical representation of average error rate

proposed method error rate is low compared to the method [34]

## 5 Conclusion

The method of estimating speed of the vehicle using geometric projection based is very accurate compared to other techniques and also promising technology for future applications in traffic monitoring. It is very powerful and cost-effective tool when it comes to dealing with traffic management techniques in the real world. The algorithm is robust even in the real time. The experimental results state that the algorithm can also compute multiple objects detection. The performance parameters were set for proposed algorithm, i.e., the sensitivity with detection rate of 96%, accuracy of about 90.8% and specificity of about 92.2%. It is a cost-effective alternative system to the traditional radar system. The future work of the system, the system can further focus more on night surveillance, also occlusion handling and there are some unexpected conditions such as the camera shake can happen and cause some errors.

**Acknowledgements** The Research is supported by Science Engineering Research Board, with File No.: SERB/SRG/2019/002277 under startup Research Grant Program in Engineering Science and is gratefully acknowledged.

## References

- Zhang, Z., Xu, H., Chao, Z., Li, X., Wang, C.: A novel vehicle reversing speed control based on obstacle detection and sparse representation. *IEEE Trans. Intell. Transp. Syst.* **16**(3), 1321–1334 (2015)
- Anandhalli, M., Baligar, V.P.: A novel approach in real-time vehicle detection and tracking using Raspberry Pi. *Alex. Eng. J.* **57**(3), 1597–1607 (2018). (ISSN:1110-0168)
- Anandhalli, M., Baligar, V.P.: An approach to detect vehicles in multiple climatic conditions using the corner point approach. *J. Intell. Syst.* **27**(3), 363–376 (2018)
- Yoon, Y., Kosaka, A., Kak, A.C.: A new Kalman–Filter-based framework for fast and accurate visual tracking of rigid objects. *IEEE Trans. Robot.* **24**(5), 1238–1251 (2008)
- Zhang, R., Li, X., Zhang, H., Jiao, Z.: Geodesic multi-class SVM with stiefel manifold embedding. *IEEE Trans. Pattern Anal. Mach. Intell.* (2021). <https://doi.org/10.1109/TPAMI.2021.3069498>
- Zhang, R., Zhang, H., Li, X.: Robust multi-task learning with flexible manifold constraint. *IEEE Trans. Pattern Anal. Mach. Intell.* **43**(6), 2150–2157 (2021). <https://doi.org/10.1109/TPAMI.2020.3007637>
- Zhang, R., Nie, F., Li, X.: Feature selection with multi-view data: a survey. *Inf. Fus.* **50**, 158–167 (2019). <https://doi.org/10.1016/j.inffus.2018.11.019>. (ISSN:1566-2535)
- Zhang, R., Tong, H.: Robust principal component analysis with adaptive neighbors. In: *NeuIPS* <https://par.nsf.gov/biblio/10159297>
- Zelinsky, A., Barnes, N., Fletcher, L.S.: Real-time speed sign detection using the radial symmetry detector. *IEEE Trans. Intell. Transp. Syst.* **9**(2), 322–332 (2008)
- Nassu, B.T., Luvizon, D.C., Minetto, R.: A video-based system for vehicle speed measurement in urban roadways. *IEEE Trans. Intell. Transp. Syst.* **18**(6), 1393–1404 (2017)
- Shamma, J.S., Odat, E., Claudel, C.: Vehicle classification and speed estimation using combined passive infrared/ultrasonic sensors. *IEEE Trans. Intell. Transp. Syst.* **19**(5), 1593–1606 (2018)
- Cao, W., Yuan, J., He, Z., Zhang, Z., He, Z.: Fast deep neural networks with knowledge guided training and predicted regions of interests for real-time video object detection. *IEEE Access* **6**, 8990–8999 (2018)
- Wong, C., Siu, W., Jennings, P., Barnes, S., Fong, B.: A smart moving vehicle detection system using motion vectors and generic line features. *IEEE Trans. Consum. Electron.* **61**(3), 384–392 (2015)
- Kosaka, N., Ohashi, G.: Vision-based nighttime vehicle detection using CenSurE and SVM. *IEEE Trans. Intell. Transp. Syst.* **16**(5), 2599–2608 (2015)
- Wu, J., Liu, Z., Li, J., Gu, C., Si, M., Tan, F.: An algorithm for automatic vehicle speed detection using video camera. In: 2009 4th International Conference on Computer Science Education, pp. 193–196 (2009)
- Zhang, R., Nie, F., Li, X.: Feature selection under regularized orthogonal least square regression with optimal scaling. *Neurocomputing* (2017). <https://doi.org/10.1016/j.neucom.2017.07.064>
- Li, X., Zhang, H., Zhang, R., Nie, F.: Discriminative and uncorrelated feature selection with constrained spectral analysis in unsupervised learning. *IEEE Trans. Image Process.* **29**, 2139–2149 (2020). <https://doi.org/10.1109/TIP.2019.2947776>
- Zhang, R., Li, X.: Unsupervised feature selection via data reconstruction and side information. *IEEE Trans. Image Process.* **29**, 8097–8106 (2020). <https://doi.org/10.1109/TIP.2020.3011253>
- Ki, Y.K., Baik, D.K.: Vehicle classification model for loop detectors using neural networks. *Transp. Res. Record* **1917**(1), 164–172 (2005). <https://doi.org/10.1177/0361198105191700118>
- Jing-Zhong, W., Xiaoqing, X.: A real-time detection of vehicle's speed based on vision principle and differential detection. In: 2009 IEEE/INFORMS International Conference on Service Operations, Logistics and Informatics, pp. 493–496 (2009)
- Rahim, H.A., Ahmad, R.B., Zain, A. S. M., Sheikh, U. U.: An adapted point based tracking for vehicle speed estimation in linear spacing. In: International Conference on Computer and Communication Engineering (ICCCE'W), pp. 1–4 (2010)
- Alberola, J., Pelegri, J., Llario, V.: Vehicle detection and car speed monitoring system using GMR magnetic sensors. In: IEEE 2002 28th Annual Conference of the Industrial Electronics Society, IECON 02, vol. 2, pp. 1693–1695 (2002)
- Pornpanomchai, C., Kongkittisan, K.: Vehicle speed detection system. In: 2009 IEEE International Conference on Signal and Image Processing Applications, pp. 135–139 (2009)
- Kumar, T., Kushwaha, D.S.: An efficient approach for detection and speed estimation of moving vehicles. *Procedia Comput. Sci.* **89**, 726–731 (2016)
- Javadi, S., Dahl, M., Pettersson, M.I.: Vehicle speed measurement model for video-based systems. *Comput. Electr. Eng.* **76**, 238–248 (2019)
- Lu, S., Wang, Y., Song, H.: A high accurate vehicle speed estimation method. *Soft Comput.* **24**, 1283–1291 (2020)
- Baek, J., Kim, J., Kim, E.: A novel on-road vehicle detection method using 7rHOG. *IEEE Trans. Intell. Transp. Syst.* **16**(6), 3414–3429 (2015)
- Song, W., et al.: Real-time obstacles detection and status classification for collision warning in a vehicle active safety system. *IEEE Trans. Intell. Transp. Syst.* **19**(3), 758–773 (2018)
- Cao, X., et al.: Vehicle detection and motion analysis in low-altitude airborne video under urban environment. *IEEE Trans. Circuits Syst. Video Technol.* **21**(10), 1522–1533 (2011)
- Hinz, S., Leitloff, J., Stilla, U.: Vehicle detection in very high resolution satellite images of city areas. *IEEE Trans. Geo-Sci. Remote Sens.* **48**(7), 2795–2806 (2010)
- Anandhalli, M., Baligar, V.P.: Vehicle detection and tracking based on color feature. In: 2017 International Conference on Recent Advances in Electronics and Communication Technology (ICRAECT), pp. 240–248 (2017)
- Stauffer, C., Grimson, W.E.L.: Adaptive background mixture models for real-time tracking. In: Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No. PR00149), vol. 2, pp. 246–252 (1999)
- Stauffer, C., Grimson, W.E.L.: Learning patterns of activity using real-time tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(8), 747–757 (2000)
- Nguyen, T.T., et al.: Compensating background for noise due to camera vibration in uncalibrated-camera-based vehicle speed measurement system. *IEEE Trans. Veh. Technol.* **60**(1), 30–43 (2011). <https://doi.org/10.1109/TVT.2010.2096832>
- Celik, T., Kusetogullari, H.: Solar-powered automated road surveillance system for speed violation detection. *IEEE Trans. Ind. Electron.* **57**(9), 3216–3227 (2010). <https://doi.org/10.1109/TIE.2009.2038395>
- Javadi, S., Dahl, M., Pettersson, M.I.: Vehicle speed measurement model for video-based systems. *Comput. Electr. Eng.* **76**, 238–248 (2019). <https://doi.org/10.1016/j.compeleceng.2019.04.001>. (ISSN:0045-7906)

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



**Dr. Mallikarjun Anandhalli** is currently working as an Associate Professor in Department of Electronics & Communication Engineering, KLS GIT, Belagavi, he pursued his Ph.D. degree in the Department of Computer Science Engineering, from VTU, Belagavi. He has completed his B. E (Electronics & Communication Engineering,) in 2011, followed by M.Tech (Digital Electronics) in 2013 and Ph.D. in 2018. He has 4 years of research experience and 4 years of Academic experience. He

has received several sponsored research projects from Government of India and Government of Karnataka. He is also currently working as a Technical Advisor to M/s Falca E Solution Pvt Ltd, Bangalore. He is associated with Medical based application related to dental problems. He is a principal investigator for central government sponsored project to address real time traffic issues. His research interest includes Computer Vision, Image and Video Processing, Medical Imaging, Machine Learning etc. He has published numerous articles in reputed peer reviewed Journals/Book/Book Chapter/Conference in the area of Image and Video processing. E-mail: malliarjun71@gmail.com.



**Pavana Baligar** received a B.E. degree in Computer Science Engineering from SKSVM. Agadi College of Engineering & Technology, India in 2013 and an M. Tech degree in Computer Science at AMC College of Engineering, Bangalore in 2015. Her research interests are Speech Emotion Recognition, Image Processing, Machine Learning and Deep Learning.



**Santosh S. Saraf** received the Ph.D. degree in Electrical and Electronics Engineering Sciences in 2018 from Visvesvaraya Technological University, India. He is currently working as Professor and Head of Department of Electronics and Communication Engineering, KLS Gogte Institute of Technology, Belagavi, Karnataka, India. Dr. Santosh's principal research interest is application of Soft Computing in the area of Multimedia Processing. He has reviewed papers for Elsevier's

Computers in Biology and Medicine, Journal of Medical Imaging and Health Informatics. Dr. Santosh is leading the Pre-Incubation Center at the Gogte Institute of Technology promoting innovation amongst students and faculty.



**Pooja Shrishail Deepsir** is currently working as an Assistant System Engineer at TATA CONSULTANCY SERVICES, BANGAL ORE. She completed her B.E in the Department of Electronics and Communication Engineering in 20 20, from KLS GIT, Belagavi. Her research interests are Vehicle speed measurement in Video, Image Processing and Machine Learning. Email: deepsir@gmail.com.