

Vehicle Tracing and Counting Structure

M. Tech. Student Aashish Meena, Assistant Professor Megha Jat

Department of Computer Science
Patel College of Science & Technology, Indore, India
meenaashish96@gmail.com, Gmegha424@gmail.com

Abstract- Traffic surveillance systems are interesting to many researchers to improve the traffic control and reduce the risk caused by accidents. In this area, many published works are only concerned about vehicle detection in normal conditions. The camera may vibrate due to wind or bridge movement. Detection and tracking of vehicles are a very difficult task when we have bad weather conditions in winter (snowy, rainy, windy, etc.) or dusty weather in arid and semiarid regions or at night, among others. In this paper, we proposed a method to track and count vehicles in dusty weather with a vibrating camera. For this purpose, we used a background subtraction-based strategy mixed with extra processing to segment vehicles. In this paper, the extra processing included the analysis of the headlight size, location, and area. In our work, tracking was done between consecutive frames via a particle filter to detect the vehicle and pair the headlights using the connected component analysis. So, vehicle counting was performed based on the pairing result. Our proposed method was tested on several video surveillance records in different conditions such as in dusty or foggy weather, with a vibrating camera, and on roads with medium-level traffic volumes. The results showed that the proposed method performed better than other previously published methods, including the Kalman filter or Gaussian model, in different traffic conditions.

Keywords- Vehicles Detection; Vehicles Tracking; Vehicle Counting; Gaussian model; foggy weather; dusty dusty.

I. INTRODUCTION

Today, many researchers are attracted to traffic because it has become a significant problem in our life. Traffic contains useful information such as site selection and engineering which can be used for different purposes. Due to the rapid developments in multimedia and wireless communication, video plays a main role in traffic management. In traffic engineering, the detection and tracking of vehicles potentially allow for traffic flow analysis, incident detection and reporting, and automated solutions to queue management. Different vehicle monitoring systems have been developed that use video in intelligent traffic systems. Besides the different advantages of the video-based method, a huge digital processing power to extract the necessary information from the image data [1, 2] can be considered as the main disadvantage of such systems.

Counting is a basic function used in many different approaches to detect and count the number of vehicles [3]. Most of the methods in this area have focused on two challenging parts of moving vehicle segmentation and recount cancellation strategy in different conditions such as high occlusion, cluttered background, dusty weather, and vibrating camera. A vehicle detection approach was presented by Cao and his colleagues [3] in which vehicles are detected in static images using color and edge. In their work, the vehicles were first extracted from the background only based on color and corner, edge maps,

and wavelet coefficient by a cascade multichannel classifier to verify possible candidates. Another automatic vehicle system classification was presented by Ilyas et al. [1] that operates based on pixel-wise relations in a region. They used edge detection to calculate local features of the image and color conversion to segment the vehicle. Also, they used a Dynamic Bayesian Network for their classification. Vehicle detection based on high vertical symmetry was developed by Litzenberger et al. [2] to facilitate vehicle detection on a video stream.

Also, Sun and his colleagues [4] proposed a new method to detect vehicles based on feature extraction and classification process. They used the Gabor filter to extract features and used support vector machine for classification. Meshram and his colleague in [5] first constructed an initial background image according to the real-time situation of traffic environment and then segmented the current frame into a foreground region and background region accurately using the combined method of interframe difference and subtraction method. Their method can detect moving vehicles fast and accurately in complex traffic situations.

Also, [6] used the connected component labeling technique for complex conditions. Since reliable vehicle detection and tracking are a critical problem for most of the existing methods, we decided to present a new system to count vehicles even in dusty weather, vibrating camera, occlusion, and background noise. These conditions are

hard to deal with, and not many works have been published working in such situation.

Zhang et al. [7] proposed a video-based vehicle detection and classification system for vehicle counting, operating under 3 different conditions: normal weather, heavy shadow in the images, and light rain with slight camera vibration. Even though their results are quite promising, their system cannot handle longitudinal vehicle occlusions, severe camera vibrations, and headlight reflection problems.

Ikoma et al. [8] proposed a method for car tracking based on bicycle specific motions in vertical vibration and angular variation via prediction and likelihood models, using particle filter for state estimation. The method was tested only under normal weather condition observing that the estimation was limited as the lighting conditions were reduced. Finally, Afo-labi et al. [9] used monocular cameras, mounted on moving vehicles such as quadcopters or similar unmanned aerial vehicles (UAVs). These cameras are subjected to vibration due to the constant movement experienced by these vehicles and consequently the captured images are often distorted.

Their approach used the Hough transform for ground line detection under normal weather conditions and concentrated on reducing the effect of the camera vibration. Our approach outperforms this point thanks to the implementation of our particle filter. In our proposed system, the vehicles entering the scene are detected and tracked throughout the video.

The input video is first fed to the algorithm; then, background subtraction is performed based on the video frames. The proposed algorithm uses this step to distinguish the vehicles from the surroundings. In the next stage, a neighborhood analysis is used to group some detected parts in the scene. After that, filtering and tracking are performed using a particle filter until the vehicles leave the scene. The remainder of this paper is organized as follows. In the next section, we will describe our proposed method and different parts of our system. Our experimental results and comparison will be mentioned in Sections 3 to 6. Finally, some concluding remarks are made in Section 7.

II. SYSTEM OVERVIEW

As depicted in Figure 1, our proposed system comprises three main components: a standard background subtraction algorithm [1] which is mixed with an extra processing module, a particle filter-based tracker, and a decision-making block.

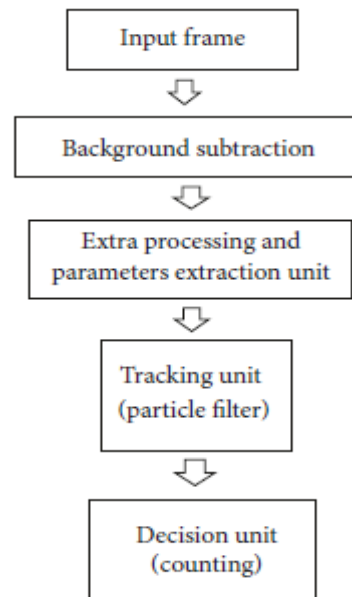


Figure 1. Flowchart of our proposed algorithm.

All these three parts are combined into a tightly coupled framework. For each input frame, the background subtraction algorithm segments the areas, and each of them is assumed to be either a vehicle or a group of vehicles. Also, since we considered videos recorded in unfavorable conditions such as dusty weather and vibrating camera, we used an extra processing unit to enhance the accuracy of our vehicle segmentation algorithm and provide better parameters for our tracking unit. For this purpose, we analyzed the headlight size, location, and area for regions resulting from the background subtraction algorithm. After that, we used a particle filter to track and find each vehicle in the next frame. Our tracker maintained the trajectory of each vehicle over time whose parameters were sent to the decision-making module. Finally, these trajectories were used in the vehicle counting applications. In the following, we will describe these three parts of our algorithm in more detail.

III. BACKGROUND SUBTRACTION AND EXTRA PROCESSING

Our proposed method can be used even if the camera is vibrating; hence, the background seen in the camera images is nearly dynamic. Since the next stages of our algorithm could be affected by the result of this section, a robust and accurate algorithm was desired to perform our background subtraction section. Our approach computed a model for the background and updated it continuously over the video scenes. For this section, we used the algorithm in [10] due to its robustness and high detection performance. The background of the videos is assumed to be static or of little movement at least between consecutive frames due to the small-time interval between

them. This approach first produced a model based on the statistical histogram of the background for each pixel and collected the color cooccurrences to increase.

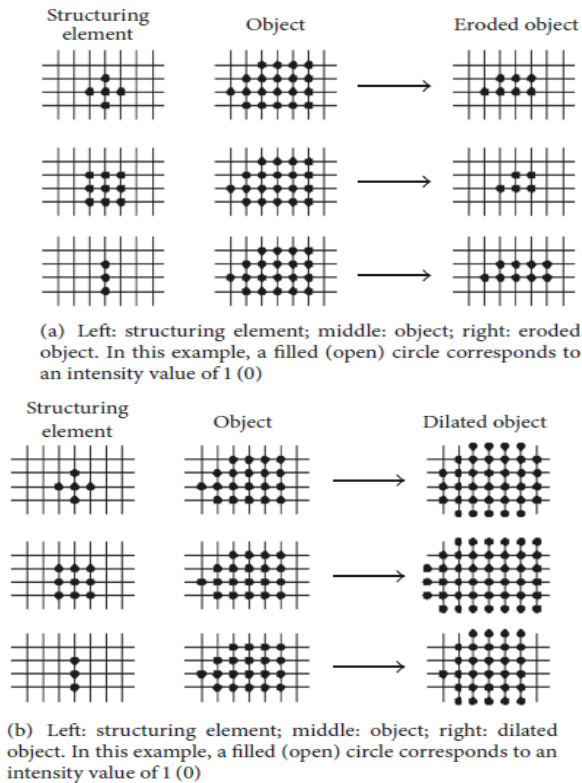


Figure 2. Erosion and dilation effects.

the accuracy of the dynamic background and classified the background. To adapt the model more, the algorithm updated the model in each frame using a weighted average filter. For this purpose, we calculate the average value and standard deviation ratio in each frame and update the previous values based on a forget factor. Also, in this paper, extra processing was used to tackle the problem of hollow phenomenon.

For this purpose, the headlight size, location, and area were analyzed and morphological operations, erosion, and dilation were used based on a disk mask as shown in Figure 2. The sample results from background subtraction in different weather condition is illustrated in Figure 3. Since we face different blobs in the results obtained from the background subtraction step and not all of them are related to vehicles and may result from the dusty weather with vibrating camera, extra processing plays a key role in our final results.

Therefore, relevant pixels were grouped and the headlight size, location, and area of blobs were analyzed to remove irrelevant blobs and reduce false detection. In order to count the number of vehicles in each frame, we needed to verify the vehicles based on their information in the last frames. For this purpose, we used a particle filter and

initialized its particles by a group of data corresponding to each vehicle and verified in the last stage. Each group of data was extracted from the region within the bounding box of each verified blob in the last stage and based on the value of the pixels. Also, a connected component analysis besides our tracking procedure could be useful to refine our search in current frame.

IV. TRACKING BY PARTICLE FILTER

Vehicle tracking is a powerful tool in traffic monitoring. Most Computer Vision algorithms applied to traffic require firstly detecting a possible vehicle and secondly segmenting it completely (i.e., obtaining its surrounding contour) and then tracking it throughout the monitored area of the image. Usually, the segmentation algorithms fail to completely obtain the surrounding contour and provide, as output, small and independent blobs of the vehicle that, put together, provide the complete shape.

Tracking allows us then to (1) intelligently combine the independent blobs that move close to each other and almost rigidly, to realize that they are part of the same vehicle; (2) help solve possible problems of occlusions due to the perspective of the camera; and (3) determine an estimation of the speed of the vehicle, once the field of view has been spatially calibrated and the information of the frame rate of the camera is known. Many works have already been published in the field of traffic monitoring, making use of recursive state estimate filters, such as the Kalman filter. This filter works well under the assumption of Gaussian distributed state variables and linear state equations. However, these conditions are not fulfilled when working with vibrating cameras, in which the dynamics are almost random, and the state equations are difficult to model.

This limitation can be overcome through the use of the particle filter, which does not assume any specific movement and provides good flexibility to obtain good tracking between frames (as long as the selected features are properly selected). To the best of our knowledge, there are no previous works using Kalman filtering and vibrating cameras in bad weather conditions. A good proposal for car tracking using Kalman filtering was made by Li et al. [11]. The authors proposed a new method relying on Kalman filtering to predict what they call the sheltered car moving position.

However, the tracking efficiency still admits some improvement. More recently, the particle filter has grown in popularity due to its flexibility and performance. Chan et al. [12] presented a method to detect and track the vehicle under various lighting conditions. The authors achieved this goal by generating a probability distribution of vehicles, using a particle filter framework, through the processes of initial sampling, propagation, observation, cue fusion, and evaluation. The detection rate of their

method under normal weather condition is 99.37%, the situation of the camera is fixed, and they did not test the method under complicated weather conditions (dusty, snowy, night, etc.). In this part of our proposed method, we tracked each verified blob to refine our counting in the next frame. In this application, we used a particle filter to iteratively generate the density of the pixels corresponding to the bounding box of the vehicles based on the Bayes rule and under the Markov assumptions.



Figure 3. Result of background subtraction algorithm.

Based on the Bayes rule, a Bayesian filter maximizes the posterior equation which is defined as follows:

$$P\left(\frac{x_t}{z_t}\right) = \eta P\left(\frac{z_t}{x_t}\right) \int P\left(\frac{x_t}{x_{t-1}}\right) P\left(\frac{x_{t-1}}{z_{t-1}}\right) \dots \quad (1)$$

Our posterior equation, $P\left(\frac{x_t}{z_t}\right)$, was defined based on the conditional probability of x_t given $Z_t = Z_{t-1} \cup Z_t$; and $Z_{t-1} = \{z_1, \dots, z_{t-1}\}$ are the measurement at t and the history of measurement until $t-1$, respectively. η is a constant which can participate in the normalization process. Although an iterative procedure can be used to derive the current state of (1), from the perspective of the particle filter, the particles could be used to achieve the best distribution based on the Bayes rule. In this framework, a particle can be encoded using a state vector and a relevant weight to define a hypothesis and its confidence. Also, resampling and generation are considered based on drawing particles following their corresponding weights at a previous time; hence, a heavier weight indicates the higher probability of a particle. Although our estimation could converge to the best posterior density function using a large number of particles, we used less than 1000 particles due to the computational complexity. In the particle filter framework, the final state vectors and weights are considered as our final state distribution.



Figure 4. Parameters definition.

In this work, we used a representative vector to describe each vehicle as follows:

$$(u_t, v_t, w_t, h_t, \Delta u_t, \Delta v_t, \Delta w_t)^t$$

u_t, v_t : coordinate of vehicle
 w_t, h_t : width and height of vehicle

$$\Delta u_t, \Delta v_t, \Delta w_t = u_t - u_{t-1}, v_t - v_{t-1}, w_t - w_{t-1} \quad \text{-----}(2)$$

The parameters of x_t are extracted and evaluated as depicted in the following sample frame for a vehicle in Figure 4. In our research, the parameters were defined as $z_t = (z_t^{im}, z_t^{un}, z_t^{ve}, z_t^{tl})^t$, including the image z_t^{im} , map of vertical edge z_t^{ve} , map of underneath shadow z_t^{un} , and map of taillight z_t^{tl} . In our definition of the parameters, we used the Sobel edge detector mixed with an orientation constraint for the map of vertical edges. We also used horizontal orientation constraints mixed with an intensity threshold for the map of the underneath shadow and used thresholds as [13] for the taillight map. After that, we generated a set of particles at $t-1$, X_{t-1} , by assigning a weight to each particle as follows:

$$X_{t-1} = \{\langle x_{t-1}^1, w_{t-1}^1 \rangle, \langle x_{t-1}^2, w_{t-1}^2 \rangle, \dots, \langle x_{t-1}^n, w_{t-1}^n \rangle\} \quad \text{---}(3)$$

where each term indicates the k th particle at $t-1$ based on state vector of $x_{(t-1)}^k$ and its weight $w_{(t-1)}^k$. Also, n is the total number of particles. There is also the following rule:

$$1 \leq k < j \leq n \rightarrow w_{t-1}^k \geq w_{t-1}^j, \forall k, j \in N \quad \text{-----}(4)$$

Propagation of the particles is one of the main stages of the particle filter. In this stage which contains two sections, deterministic drift and randomized diffusion, particles propagate to new positions in the form of guests. The propagation stage can be defined as follows:

$$\tilde{x}_t = x_t + N(0, \Sigma) \quad \text{-----}(5)$$

where x_t is the deterministic drift defined by (6), $N(0, \Sigma)$ is a random term with normal distribution (zero mean and variance of Σ).

$$x_t = \left(u_{t-1} + \Delta u_{t-1}, v_{t-1} + \Delta v_{t-1}, w_{t-1} + \Delta w_{t-1}, h_t + \Delta w_{t-1} \frac{h_{t-1}}{w_{t-1}}, \Delta u_{t-1}, \Delta v_{t-1}, \Delta w_{t-1} \right)^t$$

Hence, we can estimate the positions of the vehicles in the current frame based on drifting and diffusing of the particles estimated in the last frame. In this application, the motion model which includes noise is defined by

$$x_{t+1} = Ax_t + B_u u_t + B_f f_t \quad \text{-----}(7)$$

$$y_t = h(x_t) + n_t$$

where x_t is the state vector, u_t is measured inputs, f_t is faults, y_t is measurements, and n_t is the noise model? Although in different works a Gaussian distribution is preferred for noise model, in this work, we used a more realistic noise model to develop tracking. In our approach, noise model is specified by two parts as shown below. One part of the noise model is given for density measurements, and a second part of the noise model is given for speed measurements.

$$P(n) = \sum_{n_s=0}^{\infty} \frac{\lambda_1^{(n+n_s)} e^{-\lambda_1}}{(n+n_s)!} \times \frac{\lambda_2^{n_s} e^{-\lambda_2}}{n_s!} \quad \text{-----(8)}$$

In this work, based on noise area consideration, we assumed that the noise speed, ns , is Gaussian and $\lambda_1 + \lambda_2 = 2$, $\lambda_1 = 4/3$, $\lambda_2 = 2/3$. Also, we assumed independent distributions for f_t , n_t , and x , with known probability densities that were not necessarily Gaussian.

V. DECISION-MAKING

After these stages, we used (9) to make a decision about the vehicles and count them. For this purpose, we compared the estimated locations, $P_{t-1}^{Estimated}$, with detected blobs of vehicles, $P_t^{detected}$, in the current frame.

$$Final_{result} = \begin{cases} 1 & \|P_t^{detected} - P_{t-1}^{Estimated}\| < \delta \\ 0 & \text{else,} \end{cases}$$

VI. EXPERIMENTAL RESULTS

The proposed method was implemented in Matlab framework. Our method could process around 8 frames per second on a dual core processor at 2.4GHz. We used video datasets from highways which had unfavorable conditions such as

Table 1: Detection rate of our proposed method.

Category	correct	Missed	Detection rate (%)
Normal	450	5	98.9
Dusty	654	152	81.1
Vibrated camera	245	657	9.8
snowy	236	478	3.4

Table 2: Detection rate of Kalman based method.

Category	correct	Missed	Detection rate (%)
Normal	452	3	99.3
Dusty	589	217	73
Vibrated camera	87	120	61
snowy	195	88	69

dusty weather with a vibrating camera. For fair evaluation of our proposed method, we used test videos in different

conditions (Figure 5). These conditions were bright night and dusty and snowy weather with a vibrating camera. To evaluate the performance of the system, we used test video data collected under various weather and lighting conditions, over a period of more than seven days. The dataset made contains video data collected at different areas in Iran and under different conditions, such as vibrating camera, rainy, snowy, dusty, or foggy weather, and collisions. We collected more than 50 videos, with up to 5-minute length each.

These videos contain up to 2100 frames; the initial frames, nearly 20 percent, do not contain any vehicles. Figure 5 shows those videos corresponding to the unfavorable weather conditions. Even though the videos gather different fields of view, our system is able to monitor an area of, approximately, 80×12 meters. With this field of view, we are able to detect and track (assuming no big occlusions due to big trucks and other special vehicles) simultaneously the set of vehicles present in the monitored area, which may rise to 30 as maximum, more or less. Figure 6 shows some of the experimental results including snowy weather (snow in the air and on the ground) with vibrating camera conditions due to strong wind (which makes the picture unclear). Our method worked suitably in ALL conditions. Our simulation results confirmed the better performance of our method. In our testing scenario, we used a trajectory mechanism to evaluate our method. For this purpose, we considered several situations of vehicles in our trajectory and adapted our estimations to the detected ones.

Figure 7 illustrates some of the trajectories in different conditions. In these trajectories, several situations were estimated and compared to the detected ones for $t-k$ frames as below (red: estimated; blue: detected). As depicted in Figure 7, our proposed method used estimation data to refine its detection results for more efficient counting. In other words, when we faced frames with occlusion between vehicles or the frames had undesirable blobs due to dusty weather with vibrating camera, our estimation results which tracked the previous frames helped us to make a decision about vehicles in the current frame. The details of the detection result of our proposed method in different conditions are shown in Table 1. In this table, "Correct" means the number of vehicles detected correctly and "Missed" means the number of missed vehicles. Based on the results of Table 1, the detection rate of the proposed method was up to 98.9% in normal conditions.

However, our performance was worse when the weather was dusty and the camera was vibrating. Although our strategy is new in unfavorable conditions, we used a detection and counting method based on the Kalman filter [13] to compare the performance of our proposed method.

This algorithm is well known and is referred to by many researchers in similar works [13, 14].

The detection rates of this method in similar conditions are mentioned in Table 2.

VII. CONCLUSION AND FUTURE WORK

In this paper, we proposed a method to improve the detection and counting of vehicles in dusty weather with vibrating camera conditions. Our method uses an improved background subtraction algorithm to detect vehicles. Our innovation in this section includes a series of processing combined with a background subtraction algorithm to remove virtual blobs and decrease the effect of dusty weather with a vibrating camera. Additionally, we used a tracking step to achieve more data for the verification of vehicles in each frame based on their information on previous frames.

This idea was implemented by a particle filter and improved our detection and counting procedure. Our results showed that our proposed method was robust in complex conditions such as bright nights and snowy and dusty weather with a vibrating camera (due to the wide range of traffic conditions and varying speeds in different lanes). Our results are comparable to other works, such as those that use the Kalman filter in normal conditions, but are much better under extreme conditions, such as dusty or snowy weather with a vibrating camera. However, some improvements may still be included, as the detection rate may still be enhanced. We have detected that white cars are hard to detect under snowing conditions.

This issue requires some further research with more sensitive motion detection algorithms to be able to identify small changes on the monitored road. Additionally, the background subtraction method may be affected by vibrations of the camera or background movement due to windy conditions.

Due to that, we plan to enhance our system by extending the particle filter functionality, so that the background subtraction method may be partially or completely replaced with the particle filter. The provided accurate tracking information allows our system to face the next step, which is to measure the average speed of the vehicle in the monitored field of view. To achieve that, we will include, as future work, a 2D to 3D field of view calibration to the sequences, with continuous update of the calibration phase to take into account the dusty weather, vibration of the camera, and its effects on the extrinsic parameter of the calibration matrix.



(a). Foggy weather and bad lighting condition; resolution: 364 * 470; 187 frames.



(b). Foggy and snowy weather; resolution: 364*470; 190 frames.



(c). Foggy weather and occlusion; resolution: 364 * 470; 197 frames.



(d). Foggy weather with vibrating camera; resolution: 364 * 470; 186 frames.



(e). Foggy and rainy weather with vibrating camera and bad lighting condition; resolution: 364 * 470; 201 frames.



(f). Snowy weather with vibrating camera; resolution: 364 * 470; 185 frames.



(g). Foggy weather with vibrating camera and occlusion; resolution: 364 * 470; 186 frames.

Figure 5. Samples of datasets.



(a). Frame 408



(b). Frame 453



(c). Frame 470

Figure 6. Simultaneous tracking of vehicles, including snowy weather (snow in the air and on the ground) with vibrating camera conditions due to strong wind.

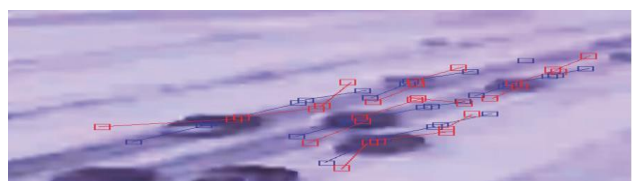


Figure 7. Evaluation in trajectory scenario.

Additionally, it would be convenient to include some information that would allow us to quantify the blurring level of the images, according to the amount of snow, dust, or fog captured by the camera. To achieve that, we have plans to incorporate an opacimeter in the monitored area, so that we can characterize the level of noise present in the images and, with that, provide estimated operational conditions of our proposal.

REFERENCES

- [1] A. Ilyas, M. Scuturici, and S. Miguet, "Real time foregroundbackground segmentation using a modified codebook model," in Proceedings of the 6th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS 2009, pp. 454–459, ita, September 2009.
- [2] M. Litzenberger, C. Posch, D. Bauer et al., "Embedded vision system for real-time object tracking using an asynchronous transient vision sensor," in Proceedings of the 2006 IEEE 12th Digital Signal Processing Workshop and 4th IEEE Signal Processing Education Workshop, DSPWS, pp. 173–178, usa, September 2006.
- [3] M. Cao, A. Vu, and M. Barth, "A novel omnidirectional vision sensing technique for traffic surveillance," in Proceedings of the 10th International IEEE Conference on Intelligent Transportation Systems, ITSC 2007, pp. 648–653, usa, October 2007.
- [4] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection using Gabor filters and support vectormachines," in Proceedings of the 14th International Conference on Digital Signal Processing, DSP 2002, pp. 1019–1022, grc, July 2002.
- [5] S. A. Meshram and V. B. Raskar, "Vehicle detection and tracking techniques used in moving vehicles," International Journal of Innovative Science, Engineering and Technology, vol. 2, no. 7, 2015.
- [6] J. Tripathi, K. Chaudharya, A. Joshia, and J. B. Jawaleb, "Automatic vehicle counting and classification," International Journal of Innovative and Emerging. Research in Engineering, vol. 2, no. 4, 2015.
- [7] G. Zhang, R. P. Avery, and Y. Wang, "Video-based vehicle detection and classification system for real-time traffic data collection using uncalibrated video cameras," Transportation Research Record, no. 1993, pp. 138–147, 2007.
- [8] N. Ikoma, Y. Mikami, and T. Ikenaga, "Car tracking in rear view based on bicycle specific motions in vertical vibration and angular variation via prediction and likelihood models with particle filter for rear confirmation support," in Proceedings of the 2014 World Automation Congress, WAC 2014, pp. 279–284, usa, August 2014.
- [9] D. Afolabi, K. L. Man, H.-N. Liang, S.-U. Guan, and T. Krilavičius, "Monocular line tracking for the reduction of vibration induced during image acquisition," JVE International Ltd. Journal of Vibroengineering, vol. 17, no. 2, pp. 655–661, 2015.
- [10] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection: a review," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 5, pp. 694–711, 2006.
- [11] H. Li, P. Bai, and H. Song, "Car tracking algorithm based on Kalman filter and compressive tracking," in Proceedings of the 2014 7th International Congress on Image and Signal Processing, CISP 2014, pp. 27–31, chn, October 2014.
- [12] Y. M. Chan, S. S. Huang, L. C. Fu, P. Y. Hsiao, and M. F. Lo, "Vehicle detection and tracking under various lighting conditions using a particle filter," IET Intelligent Transport Systems, vol. 6, no. 1, pp. 1–8, 2012.
- [13] C. Yang, R. Duraiswami, and L. Davis, "Fast multiple objects tracking via a hierarchical particle filter," in Proceedings of the Proceedings - 10th IEEE International Conference on Computer Vision, ICCV 2005, pp. 212–219, chn, October 2005.
- [14] W. Du and J. Piater, "Tracking by cluster analysis of feature points using a mixture particle filter," in Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance, AVSS 2005, pp. 165–170, ita, September 2005.
- [15] V. Prakaulya, N. Pareek, and U. Singh, "Network performance in IEEE 802.11 and IEEE 802.11p cluster based on VANET," in Proceedings of the International Conference on Electronics, Communication and Aerospace Technology, ICECA 2017, 2017, vol. 2017-Janua, doi: 10.1109/ICECA.2017.8212713.