Relative Vehicle Velocity Estimation Using Monocular Video Stream

Deepak Kumar Jain
Institute of Automation
Chongqing University of Posts and Telecommunications
Chongqing, China
deepak@cqupt.edu.cn

Dr. Rachna Jain
Department of Computer Science & Engineering
Bharati Vidyapeeth's College of Engineering
Delhi, India
rachna.jain@bharatividyapeeth.edu

Lingin Cai
Institute of Automation
Chongqing University of Posts and Telecommunications
Chongqing, China
iamlqcai@163.com

Meenu Gupta
Department of Computer Science & Engineering
Chandigarh University,
Punjab, India
gupta.meenu5@gmail.com

Yash Upadhyay
Department of Computer Science & Engineering
Bharati Vidyapeeth's College of Engineering
Delhi, India
yash1997@gmail.com

***Abstract*:** In the past few years, the intelligent driving systems have witnessed rapid development, either it is self-driving cars or driver assistant systems. All these systems are built around perceiving the environment of the vehicle and taking appropriate steps in the given context. Computer vision has been playing a significant role in reducing the number of costly sensors used to perceiving the environment. In the past, the velocity of the vehicle was major estimated using sensors. In this paper, we propose a data-based methodology to estimate the relative velocity of vehicles using monocular cameras hence omitting the need for costly sensors such as lidars. Our proposed methods achieve a low mean velocity square error of 1.806, for estimating the velocity of the vehicle in a real-time environment.

***Keywords* — Convolution Neural Network (CNN), Traffic Light Detection (TLD), Velocity, Advanced Driver Assistance Systems (ADAS), Optical Flow, Road Detection, Vehicle Tracking, Intelligent Transportation Systems (ITS).**

I. Introduction

Intelligent Transportation Systems (ITS) are more popular among transportation due to its advantage, such as saving human lives, time, and money. These ITS systems are helpful in finding the width of the lane, traffic density, speed of vehicle on the road, and traffic volume [15] [31]. These kinds of information provided by ITS system are helpful in controlling and managing the traffic on the road, which saves time, money and life of the human. Mainly two types of methods are discussed by which used to measure the speed of vehicle such as hardware (i.e., Ground induction coil loop detectors [32], Sensors [33]), and software based (i.e., using camera calibration).

Estimation of vehicle velocity is the main features of the Intelligent Transport System (ITS), which uses intelligent vehicle camera (i.e., 1D, 2D, and 3D) to overcome the traffic accidents and improve traffic safety. Installation of 3D camera is difficult to adjust. Most of the intelligent cameras are image-based and suitable for highways because of its simple background, but in the case of urban and suburban roads, are difficult due to its complex background [2]. The methods discussed in [17]-[24] are mainly used for detecting the front vehicle's distance. Figure 1 shows the additional intelligent camera installed in the automobile to estimate the velocity of the vehicle. In case of bad weather condition, the additional camera was placed in the vehicle to see the traffic light (range from 10m -115m) [3].

Fig 1: Intelligent camera placed in a vehicle for estimation of vehicle velocity [3]

The main objectives of this paper are to estimate relative velocity of on-road vehicles, to remove the need to calibrate every camera before deploying the model. This proposed work also helps to reduce the inference time by using feature extraction.

The paper is summarized as follows: section 2 (related work) discusses the different researchers view regarding the

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

estimation of vehicle velocity using different techniques in brief. The data description, proposed methodology, and feature extractions are discussed in section 3. Section 4 shows the experiential and its performance analysis. Last section (i.e., 5) discusses the conclusion part of this paper in detail.

II. RELATED WORK

Different researchers have given their different views in respect to the estimation of vehicle velocity using different techniques/ methods. Out of these discussions, few of them are described below:

In [1], the authors considered the data of CVPR2017 vehicle velocity estimation challenge. They used a multilayer perceptron technique to propose a lightweight approach for regressing vehicle velocities from their trajectories and study of features for monocular vehicle velocity estimation. After this, they compared their features and concluded that characteristics of light-weight trajectory perform outstandingly for depth and motion cues mined from deep ConvNets. The proposed approach (i.e., based on the real-time concept) was capable of outperforming on a single CUP for competing for the entries in the challenge of velocity estimation.

In [2], authors worked on front vehicle detection and estimated its distance intelligent vehicle video camera of single-lens on roads of urban and suburban areas. They applied Hough transform technique to detect the straight line and its intersection point that is exploited for obtaining the disappearing points, and these points are segmenting the path (i.e., road) area by using edges (i.e.,,, left and right extracted). Further, they calculated the distance between host and front vehicle by measuring the disappearing points and vehicle positions. In results, they concluded that the proposed techniques had given a 78% accuracy result for the detection of the vehicle in urban and suburban road areas.

In [3], the authors proposed a new algorithm that aimed to identify the condition of road traffic light (in both day and night) and also it estimates the distance of light using its color (i.e., LED spotlights and bulb lights) properties. The authors used a fuzzy technique to get a better division of traffic light colors. Further, they used filter rules to find the aspect ratios of lights and tracking systems for analyzing whether the spots (on frames) expected as traffic lights. Bayesian filter technique in traffic light (on frames) is applied in authors work for finding the distance between traffic lights and autonomous vehicles. The authors proposed algorithm was tested in public road urban driverless car (>1 hour of video) of Italy in 2013 with 99.4% accuracy. In result, they concluded that this proposed method is not only helpful in the urban road even with highways also.

In [4], a new method called camera calibration has been proposed for measuring the speed of vehicles on the road. In this method, camera calibration was done at adjustable positions to cover the proper image of traffic. Authors drew a 2D equilateral triangle on the ground (i.e., within the view scene of the camera as a reference object) to determine the parameters of the camera. After implementation, they concluded that this method is straightforward to install with any condition of the road.

In [5], the authors applied image sensor technology to analyse the distance between two vehicles (i.e., ITS). They compared their proposed algorithm with the existing one, which consists of two image sensors with one LED for distance estimation. As a result, the authors concluded that 10 to 100 m of two-car distance is achieved 0.1m to 1.8m in accuracy with the help of the proposed approach, which is adequate for most vehicle safety applications.

In [6], the authors implemented two methods (Distance Estimation and car position detection) in ITS for finding distance for the former vehicles based on the monocular camera. They used the Hough Transform and Kalman filters technique for the detection of lane and tracking.

In [7], the authors proposed an estimation algorithm for vehicle measurement using a monocular camera. The author's work considered two cases (i.e., detected lane information and pin-hole camera model) of vehicle width for target vehicle. Kalman filter method was used to measure the acceleration, velocity, and time to the collision of target vehicles. They concluded that the proposed estimation method for output measurement could improve the performance of the FCW (Forward Collision Warning) /AEB (Autonomous Emergency Braking) systems.

In [8], the authors used image processing to analyse the visibility condition to drivers on the road. In this, the authors used an approach which used to analyse the presence of atmospheric visibility distance. This approach has shown that the estimation range of the distant object to the plane of the road was at least 5% of the contrast by using a single camera. They concluded with three degrees of freedom (i.e., longitudinal, lateral, and yaw angle) in this approach to give visibility estimation for the original image of fog.

In [9], the authors proposed an algorithm to overcome the problem of monocular cameras (which cannot measure the distance of the object directly). Authors proposed a novel algorithm (i.e., multi-object tracking) which considers input in image form and creates paths of sensed objects in a world coordinate system. Deep Neural Network (DNN) and Poisson Multi-Bernoulli Mixture Tracking (PMBM) were applied to identify and approximate the distance to objects from a single input image. As a result, they concluded that this proposed algorithm could be able to successfully track the multiple objects (i.e., in 3D) from a single camera image, which can give helpful information for making decision and control.

In [10], they used SVM (Support Vector Machine) classifier to find the position of vehicles and HOG (Histogram of Oriented Gradient) was used to verify the shadow regions at the bottom of the vehicle. With the help of video camera of a single lens, they proposed a driver assistant system for detection of a vehicle. They measured the distance of a vehicle (in the urban and suburban areas). As a result, they achieved a 94.08% accuracy rate in measuring the performance of vehicle detection with different scenes of urban and suburban road areas.

In [11], authors have given a study (using sparse optical flow technique) to measure the speed of the vehicle for a safe journey by using the monocular video camera. They used C++

programming language for their implementation, and in result, they estimated the speed of vehicle approximately ± 1-2 km/h. They worked on side view image of the road for speed estimation in their previous work [16]. In [11], they enhance their previous work of [16] by adding additional steps (worked on top view image of the road) as compared to their earlier system. In the end, they concluded that the results-driven from both the methods [11] [16] have the same accuracy.

In [12], they proposed a mixed approach of lane and vehicle detection for a Forward Collision Warning System (FCWS) that was implanted in an automatic driving system operating in real-time. This proposed framework is applicable for Advanced Driver Assistant Systems (ADAS) for a safe drive.

In [13], the authors used the 3D camera model to detect the distance and position of a vehicle by mapping the pixel position (in the form of vehicle plane and distortion parameter). They designed a framework (i.e., Extended Kalman Filter (EKF)) to track the sensed vehicle's derivative relationship between the camera and world coordinate systems. As a result, they concluded that this technique is accomplished of tracking the 3D position of the vehicle with the best accuracy as compared to the LIDAR and Radar system.

In [14], the authors considered the case of a stereo vision system for measuring the distance of vehicles. They overcome the problem of the stereo vision system (i.e., due to sampling and camera sensor error, it's challenging to measure long and short distance accurately) in this proposed approach, act as a vehicle application which works as assistance to the driving system. They used Strong Tracking Kalman Filter (STKF) to overcome the error of the camera sensor and used the sub-pixel displacement method to improve the accuracy of disparity. This proposed method is further compared with other displacement method and Conventional Kalman filter (CKF) through simulating on the several distance ranges. As a result, they have shown that accuracy of APF (Asymmetric Parabola Fitting) with the NCC (Normalized Cross-Correlation) method is superior to other methods, which are SPF (symmetric parabola fitting) with NCC and Enhanced NCC (ENCC), and Kalman filter reduces sensor noise.

In [15], a vision-based speed measurement system (for vertical and horizontal histogram) is proposed in this work. This approach was used to overcome the problem of noise that came from the displacement (consider both incoming and background image), which was caused by camera vibration over time. The authors worked on two main objectives (background compensation and automatic vanishing point detection) in this study. They concluded that the proposed method given satisfactory results in measuring the speed of the vehicle.

Different researchers have given their different views in estimating vehicle velocity and in distance measurement of vehicle. In our proposed work, we use TuSimple Velocity estimation dataset for camera calibration in a vehicle to measure the vehicle velocity estimation.

III. PROPOSED METHODOLOGY

In this proposed work, we used TuSimple Velocity estimation dataset. This proposed work uses a Neural Network (NN) technique for camera calibration. This section defines the data set used and proposed methodology implemented for measuring vehicle velocity estimation. Here, we propose a method to overcome the research gaps witnessed in the research conducted. The most common gap we found out that there are no techniques that can be used for any device regardless of the camera configurations, as methods which are dependent on the parameters which are specific for different cameras, therefore, these methods are specific for the hardware being used to collect the data. This section defines the data set used and proposed methodology implemented for measuring vehicle velocity estimation.

A. Dataset Used

Estimation of the velocity of the vehicle is mostly carried out using stereo cameras which makes it easier to estimate the depth of the objects in the images but in case of monocular cameras to convert 2-d image coordinate system to 3-d world coordinate system we need to calculate certain camera parameters which are unique for every camera. In different approaches that only deal with camera calibration there is almost no need to create datasets. Therefore, there are a limited number of publicly available datasets which can be used for developing a data-based approach for estimating velocities of the vehicle rather than calibration (every camera used for deployment). In this paper, we use the TuSimple Velocity estimation dataset, which was used for a competition of the same name in 2017.

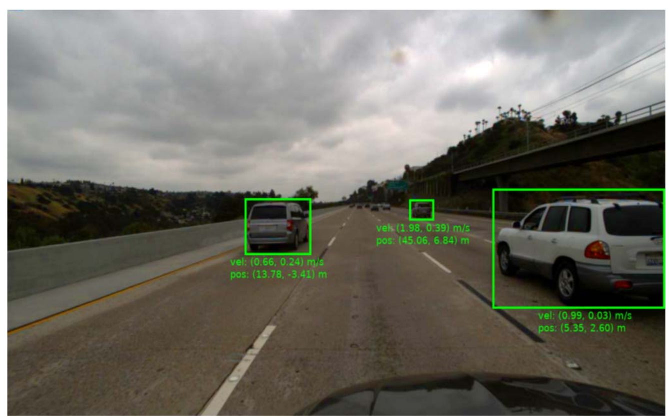

Fig 2: Training sequence sample image of the vehicle for measuring position on ground and velocity

The training dataset provides 1074 sequences of 40 frames, each collected at 20 frames per second for a duration of 2 seconds. The images are gathered in a day time environment where the relative distance between the cars ranges from 5 to 90 meters. The annotations are provided for the 40th frame of each sequence, which consists of the velocity of the vehicle along the X-axis and the Y-axis, its distance from the camera, and its position in the image defined by a bounding box.

B. *Proposed Methodology*

We present a method that takes into account both temporal and spatial features extracted from the sequences to estimate the velocity of the vehicle. The proposed method consists mainly of two steps, one is feature extraction and second in neural architecture.

The proposed method is shown by figure 3, displays that the sequences of frames that are total 40 in number will be taken from the dataset; the dense optical flow will be calculated by comparing every consecutive frame. This dense optical flow will be saved for further used meanwhile a tracker will track the cars present in the sequences which conclude the feature extraction block. Then the data extracted from the feature extraction step while be used a combination of 3 networks simultaneously, the optical flow will be used for LRCN (Long Term Recurrent Convolution Networks), and tracking will be used for MLP (Machine Language Programming) [34]-[35].

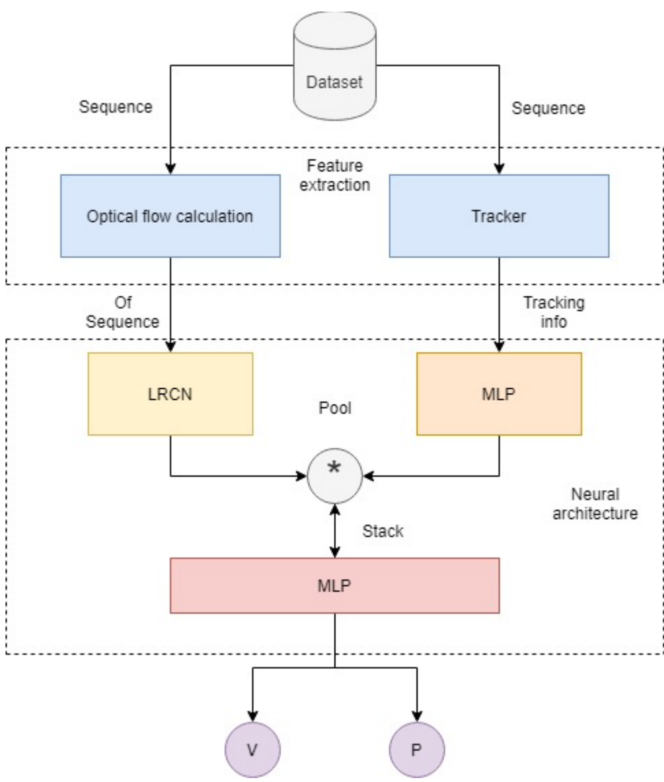

Fig 3: Proposed Methodology using MLP.

The outputs from both networks will be stacked in 1 dimension yielding a 1 X n shaped output vector, which will be used to train the third MLP, which will have two output nodes for prediction of the velocity of the vehicles present in the sequence.

C. *Features Extraction*

Although the modern-day neural architectures which are used for video analysis are capable of extracting information in both spatial and temporal domains of emphasizing the movement of the vehicles in the entire sequence of frames, we use dense optical flow [25,29] which is calculated using Farneback's algorithm. Optical flow is a well-established method of motion estimation, as it studies the relative motion between the object of interest and camera. It calculates how much a pixel moved from one frame to another, and we used OpenCV's function to calculate this flow, it returns the magnitude and the angle of the vector of displacement of the pixel. We treat this displacement of pixels as an essential feature for estimating the velocity of the vehicle. The second important feature that we are concerned is the position of the vehicle in all the frames in the sequence as the dataset only provides the position of the vehicle for the last frame we are clueless about the position of the vehicle in 1st to 39th frames for this we use DeepSort [26] vehicle tracking algorithm which uses YoloV3 for detecting vehicles and applies sort strategy to track the vehicle in all the frames. The DeepSort has been implemented using the TensorFlow 1.8 and can be easily inferred using a GPU to make the process of finding tracked bounding boxes of the vehicles faster than CPU.

D. *Neural Architecture*

The features extracted, i.e., optical flow and the tracking information of the vehicle are treated as an input for two different networks from which the extracted features are stacked to create a new input for a third network which finally predicts the value for the velocity of the vehicle and the distance of the same vehicle from the camera. For processing the optical flow, we used a long term recurrent convolutional network [27] and a 6-layer MLP for the tracking and 3-layer MLP for predicting the velocity and the position of the vehicle. The network is trained for 20 epochs using the mean square error as the loss function, and Adam optimizer [30] was used for the optimization.

IV. EXPERIMENT RESULTS

For experimental analysis, we divided the considered data into three different categories i.e., train dataset, validate the dataset, and finally test dataset. A dataset of 1074 sequence of records is considered for execution. Out of these records, 700 sequences were included in the training part, 200 for validation, and rest were taken as a testing set of sequence. For every sequence, the dense optical flow was calculated and stored as images in HSV formats. Simultaneously, DeepSort was used to track all the vehicles in the sequence and match the tracked vehicle to the annotations available. All the experimentations are performed on an Nvidia GTX 1060, which has a memory 6 GBs, for implementation Pytorch deep learning framework.

The metric used for evaluation of the results are the same as the metrics used in the TuSimple Velocity estimation challenge, the first metric used is the Mean Squared velocity error, which is defined by equation 1.

$$Ev = \frac{\sum_{c=0}^{c=C} \parallel Vgt - Vest \parallel^2}{C} \quad (1)$$

Where C denotes the submitted results for each vehicle, Vgt denotes the ground truth of the velocity of the vehicle and Vest the estimated value of the velocity by the model.

Based on the distance of the vehicles are classified as Near (0-20m), Medium(20-45m), and Far(45m+). The performance of each class is calculated separately and is averaged to get the final error, as shown in equation 2.

$$E_{final} = \frac{E_{near} + E_{med} + E_{far}}{3} \qquad (2)$$

Where E_{final} is the final velocity error that will be compared in the end, $E_{near}$is the velocity error for vehicles with a distance of 0-20 meters, E_{med} represents the error for vehicles with a distance of 20-45 meters and the $E_{far}$is for the vehicles, that is at a greater distance than 45 meters they are classified as being far.

Table 1: Experimental results

Method used	E_{final}	E_{near}	E_{med}	E_{far}
Proposed method	1.806	0.95	1.23	3.24
Tracked bounding boxes +ANN	5..61	3.64	5.67	7.54
CNN + LSTM	3.323	1.27	3.74	4.96
3d CNN	3.23	1.56	3.23	5.78

Table1 showcases the comparison between different methods tested out during experimentation, and it clearly shows that when we use optical flow features along with tracking coordinated of the vehicle, then tracked bounding boxes provide a better result. For getting a better understanding of the results, the cars have been classified into 3 categories according to the distance of the car from the camera such that it can be easily studied if the model being used is able to properly estimate the velocities of the vehicle of a different class.

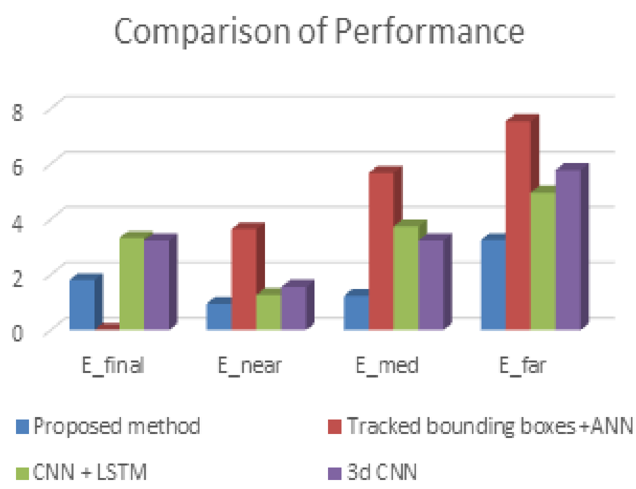

Fig 4: Graphical comparison of experimental results

In fig 4, we can see the graphical comparison of the performance of the different methods tested. The tables show that we got the lowest error for near vehicles which was 0.95 when using pur proposed method, if we trained an ANN [28] using just the tracked bounding boxes the loss increases which can be explained by the fact temporal features have not been taken into considerations and we get an Error of 3.64 for near vehicles, 5.67 for vehicles at a medium distance and 7.54 for vehicles which are far away. When employing only an LRCN, we get 1.27 for vehicles at a near distance, 3.74 for medium distance vehicles, and 4.96 losses for vehicles that are far away from the observer. When using 3d CNN [28] we get an average loss of 3.23, as it gets a loss of 1.56, 3.23, 5.78 in case of vehicles that are near, medium, or far lengths.

Table 2: Comparison between Actual and estimated velocities

S.no	Actual Velocity		Estimated Velocity	
	Vx	Vy	Vx	Vy
1	-3.2271	0.0333	-2.75	1.23
2	-0.3011	0.2174	-0.803	1.53
3	1.637	0.193	1.23	0.323
4	3.835	0.0816	2.54	0.023
5	2.627	0.041	2.13	0.03

Table 2 shows the comparison between the estimated velocities and the actual ground truth velocities for the proposed method. The velocities mentioned are relative to another moving vehicle on which the camera was situated, all velocities are mentioned in meters per second.

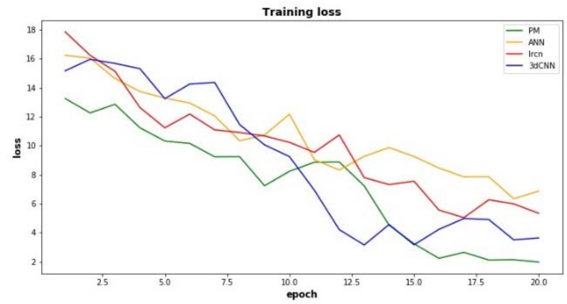

Fig 5: Training loss for experimented methods

In Fig 5, we can see the training loss for the methods that have been used for the comparison study. The loss shown is for the 20 epochs for which the network was trained for.

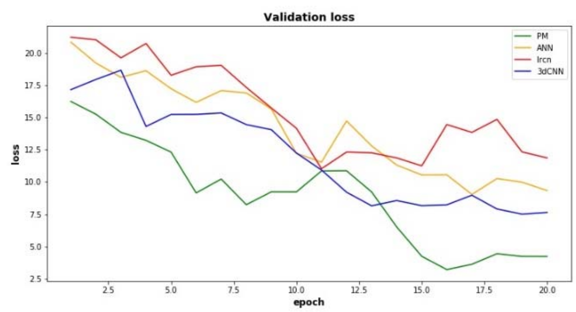

Fig 6: Validation loss for experimental methods

In Fig 6, we can see the loss curve for different methods used in the comparison study. From figure 5 and figure 6, we can conclude that our proposed method gives improved results than the other discussed methods by different researchers not only on the training set but also on the validation set. As validation and the training loss are almost identical to each other, we can conclude that our model did not overfit on the training data and can generalize over the validation data.

V. Conclusion

The estimation of velocity is important for making various decisions. This task should not only be accomplished accurately but also quickly as in a real driving environment, the time to decide in different scenarios is very less. The proposed method uses in this work is a lightweight architecture and depends on pre-calculated optical features, which are easy to calculate and can be done in real-time. Our proposed method shows the importance of optical features as well as the tracking information for estimating a velocity of a vehicle. The results of this method concluded that the proposed methods are best, and based on the calibration of the camera as it does not depend on any hardware for estimating the velocity of the vehicles. Our method of extracting particular types of feature outperforms other possible methods which do not employ feature extraction. Our method is able to outperform such methods in all the distance classifications of vehicles and also gets the lowest average error among the methods tested. Our method offers great flexibility for future improvements as the internal neural architectures can easily be replaced with new and better architectures proposed in the future. As currently there's a paucity of the annotated data for this task if new datasets are introduced then our proposed methods can be trained on those datasets to get much better accuracy scores.

References

[1] M. Kampelmühler, M. G. Müller, and C, "Feichtenhofer. Camera-based vehicle velocity estimation from monocular video", *arXiv preprint arXiv:1802.07094,* 2018.

[2] C.H.Chen, T. Y. Chen, D. Y. Huang,and K. W. Feng,"Front vehicle detection and distance estimation using single-lens video camera", In *2015 Third International Conference on Robot, Vision and Signal Processing (RVSP)* (pp. 14-17). IEEE, Nov 2015.

[3] M. Diaz-Cabrera, P. Cerri, and P. Medici,"Robust real-time traffic light detection and distance estimation using a single camera". *Expert Systems with Applications*, vol. *42, no.* 8, pp.3911-3923, 2015.

[4] V.H.Do, L.H. Nghiem, N. P. Thi, and N. P. Ngoc,"A simple camera calibration method for vehicle velocity estimation", In *2015 12th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)* (pp. 1-5). IEEE, Jun 2015.

[5] V. T. B. Tram and M. Yoo, "Vehicle to vehicle distance estimation using camera-based visible light communications", In *2017 International Conference on Information and Communication Technology Convergence (ICTC)* (pp. 517-519). IEEE, Oct 2017.

[6] A. A. Ali and H. A Hussein, "Distance estimation and vehicle position detection based on monocular camera", In *2016 Al-Sadeq International Conference on Multidisciplinary in IT and Communication Science and Applications (AIC-MITCSA)* (pp. 1-4). IEEE, May 2016.

[7] J. Han, O. Heo, M. Park, S. Kee,and M. Sunwoo, "Vehicle distance estimation using a mono-camera for FCW/AEB systems", *International journal of automotive technology*, vol. *17, no.* 3, pp.483-491, 2016.

[8] C. Boussard, N. Hautière,and B. d'Andréa-Novel, "Vision guided by vehicle dynamics for onboard estimation of the visibility range", *IFAC Proceedings Volumes*, vol. *40, no.* 15, pp.324-329, 2007.

[9] S. Scheidegger, J. Benjaminsson, E. Rosenberg, A. Krishnan, and K. Granström,"Mono-camera 3d multi-object tracking using deep learning detections and pmbm filtering", In *2018 IEEE Intelligent Vehicles Symposium (IV)* (pp. 433-440). IEEE, Jun 2018.

[10] D. Y. Huang, C. H. Chen, T. Y. Chen, W. C. Hu, and K. W. Feng, "Vehicle detection and inter-vehicle distance

estimation using single-lens video camera on urban/suburb roads", *Journal of Visual Communication and Image Representation*, vol. *46*, pp.250-259, 2017.

[11] M. S. Temiz, S. Kulur, and S. Dogan, "Real time speed estimation from monocular video", *International archives of the photogrammetry, remote sensing and spatial information sciences*, vol. *39*, p.B3, 2012.

[12] H. Kim, Y. Lee, T. Woo, and H. Kim, "Integration of vehicle and lane detection for forward collision warning system", In *2016 IEEE 6th International Conference on Consumer Electronics-Berlin (ICCE-Berlin)* (pp. 5-8). IEEE, Sept 2016.

[13] A. Kuramoto, M. A.Aldibaja, R. Yanase, J. Kameyama, K. Yoneda, and N. Suganuma, "Mono-Camera based 3D Object Tracking Strategy for Autonomous Vehicles", In *2018 IEEE Intelligent Vehicles Symposium (IV)* (pp. 459-464). IEEE, Jun 2018.

[14] Y. C. Lim, C. H. Lee, S. Kwon, and W. Y. Jung, "Distance estimation algorithm for both long and short ranges based on stereo vision system", In *2008 IEEE Intelligent Vehicles Symposium* (pp. 841-846). IEEE, Jun 2008.

[15] T. T. Nguyen, X. Dai Pham, J. H. Song, S. Jin, D. Kim, and J. W. Jeon, "Compensating background for noise due to camera vibration in uncalibrated-camera-based vehicle speed measurement system", *IEEE Transactions on Vehicular Technology*, vol. *60, no.* 1, pp.30-43, 2010.

[16] S. Doğan, M. S. Temiz, and S. Külür,"Real Time Speed Estimation Of Moving Vehicles From Side View Images From An Uncalibrated Video Camera",*Sensors,vol.* 10, no. 5, pp. 4805-4824, 2010.

[17] H. Kong, J. Y. Audibert and J. Ponce. "Vanishing Point Detection for Road Detection", *Proc. of 2009 IEEE Computer Vision and Pattern Recognition (CVPR)*, Miami, FL, USA, pp.20-25, June 2012.

[18] C. Y. Tseng and S. J. Wang, "Automatic 3-d Depth Recovery from a Single Urban-Scene Image", *Proc. of 2012 IEEE Conference on Visual Communications and Image Processing (VCIP)*, San Diego, CA, USA., pp.1-6, Nov. 2012.

[19] T. H. Bui, T. Saitoh, and E. Nobuyama, "Road Area Detection Based on Texture Orientations Estimation and Vanishing Point Detection", *Proc. of 2013 Annual Conference Society of Instrument and Control Engineers (SICE)*, Nagoya, Japan, pp. 14-17, Sep. 2013.

[20] K. Lu, J. Li, and X. An, "Road Vanishing-Point Detection: A Multi-Population Genetic Algorithm Based Approach", *Proc. of 2013 Chinese Automation Congress (CAC*), Changsha, China, pp. 415-419, Nov. 2013.

[21] H. Kong, J. Y. Audibert, and J. Ponce, "General Road Detection From a Single Image", *IEEE Transactions on Image Processing*, Vol. 19, No. 8, Aug. 2010, pp. 2211-2220.

[22] G. K. Siogkas and E. S. Dermatas, "Random-Walker Monocular Road Detection in Adverse Conditions Using Automated Spatiotemporal Seed Selection", *IEEE Intelligent Transportation Systems*, vol. 14, no. 2, pp. 527-538, Oct. 2012.

[23] C. Chun, D. Park, W. Kim, and C. Kim, "Floor Detection Based Depth Estimation from a Single Indoor Scene", *Proc. of IEEE International Conference on Image Processing (ICIP)*, Melbourne, VIC, pp. 3358-3362, Sep. 2013.

[24] B. Wang, V. Fremont, and S. A. Rodriguez, "Color-Based Road Detection and Its Evaluation on the KITTI Road Benchmark",*Proc.of 2014 IEEE Intelligent Vehicles Symposium*, Dearborn, MI, pp. 8-1, June. 2014.

[25] D. Sun, S. Roth, and M. J. Black,"Secrets of optical flow estimation and their principles", In *2010 IEEE computer society conference on computer vision and pattern recognition* (pp. 2432-2439). IEEE, June 2010.

[26] A. Bewley, Z. Ge, L. Ott, F. Ramos, andB. Upcroft, "Simple online and realtime tracking. In *2016 IEEE International Conference on Image Processing (ICIP)* (pp. 3464-3468). IEEE, Sept 2016.

[27] K. Saleh, M. Hossny, A. Hossny, andS. Nahavandi, "Cyclist detection in LIDAR scans using faster R-CNN and synthetic depth images", In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)* (pp. 1-6). IEEE, Oct 2017.

[28] S. Ji, W. Xu, M. Yang and K. Yu, "3D Convolutional Neural Networks for Human Action Recognition," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221-231, Jan. 2013. doi: 10.1109/TPAMI.2012.59

[29] A. Buades, J. Lisani and M. Miladinović, "Patch-Based Video Denoising With Optical Flow Estimation," in *IEEE Transactions on Image Processing*, vol. 25, no. 6, pp. 2573-2586, June 2016. doi: 10.1109/TIP.2016.2551639

[30] Z. Zhang, "Improved Adam Optimizer for Deep Neural Networks," *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*, Banff, AB, Canada, 2018, pp. 1-2. doi: 10.1109/IWQoS.2018.8624183

[31] P.-Y. Hsiao, C.-W. Yeh, S.-S. Huang, and L.-C. Fu, "A portable visionbasedreal-time lane departure warning system: Day and night," *IEEETrans. Veh. Technol.*, vol. 58, no. 4, pp. 2089–2094, May 2009.

[32] Y. K. Ki and D. K. Baik, "Model for accurate speed measurement using double-loop detectors," *IEEE Trans. Veh. Technol.*, vol. 55, no. 4, pp. 1094–1101, Jul. 2006.

[33] M. Litzenberger, A. N. Belbachir, N. Donath, G. Gritsch, H. Garn, B. Kohn, C. Posch, and S. Schraml, "Estimation of vehicle speed based on asynchronous data from a silicon retina optical sensor," in *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, 2006, pp. 653–658.

[34] Alzubi, J., Nayyar, A., & Kumar, A. (2018, November). Machine learning from theory to algorithms: an overview.

In Journal of physics: conference series (Vol. 1142, No. 1, p. 012012). IOP Publishing.
[35] Y. Peng, C. Jian, and M. Yan, "Observer-based estimation of velocity and tire-road friction coefficient for vehicle control systems," *Nonlinear Dynamics*, vol. *96, no.* 1, pp.363-387, Apr. 2019.