Vehicle Detection And Accident Prediction In Sand/Dust Storms

Arun Singh
Student, Integrated B. Tech M. Tech,
Computer Science and Engineering,
Lovely Professional University
Phagwara, India
arunmandiarun2001@gmail.com

Dora Praveen Kumar
Student, Integrated B. Tech M. Tech,
Computer Science and Engineering,
Lovely Professional University
Phagwara, India
kumardorapraveen@gmail.com

Kelothu Shivaprasad
Student, Integrated B. Tech M. Tech,
Computer Science and Engineering,
Lovely Professional University
Phagwara, India
keloth.shivaprasad@gmail.com

Mohit
Student, Integrated B. Tech M. Tech,
Computer Science and Engineering,
Lovely Professional University
Phagwara, India
mohitmalik4231@gmail.com

Ankita Wadhawan
Assistant Professor
Computer Science and Engineering,
Lovely Professional University
Phagwara, India
ankita.23891@lpu.co.in

***Abstract*—In this era of a smart and modern world that is designed by progressing technology, automated vehicles would become a precious part of it. The first thing that strikes in our minds talking about vehicles is traffic and accidents. Accidents could take place because of several reasons: dense traffic, unfavorable weather conditions, sudden braking, change in speed, etc, and the solution to this is machine learning, computer vision, and deep learning. Our focus is to improve the vision in areas of low visibility and predict the future by analyzing the present. Here we introduce a model which would help in dehazing and improving the visibility for a better driving experience in adverse weather especially targeting sandstorms and dust storms which would be quite common in the future because of the afforestation, the procedure is divided into two categories the dehazing and second is vehicle detection, situation analysis, and prediction. We have also incorporated things like estimating traffic density(dense/sparse), and the fire's in the worst situation using python, tensor flow, deep learning, and counting vehicles entering and departing from the frame.**

***Keywords*—**traffic surveillance, computer vision, automatic car detection, transportation systems, image pre-processing, sand/dust storms.

I. Introduction

Vehicle recognition and following assume a significant part in self-ruling vehicles and keen transportation systems. Unfriendly climate conditions like the presence of hefty day off, downpour, residue, or dust storm circumstances are hazardous limitations on a camera's capacity by decreasing perceivability, influencing driving security.

These limitations sway the exhibition of identification and following calculations used in the rush hour gridlock reconnaissance frameworks and independent driving applications. In this article, we start by proposing a perceivability upgrade plot comprising three phases: brightening improvement, reflection part upgrade, and straight weighted combination to improve the exhibition. At that point, we present a strong vehicle identification and the following methodology utilizing a multi-scale profound convolution neural organization. The regular Gaussian combination likelihood theory thickness channel-based tracker is used mutually with progressive information affiliations (HDA), which parts into recognition-to-track and track-to-follow affiliations. In this, the expense lattice of each stage is settled utilizing the Hungarian calculation to make up for the lost tracks brought about by missed identification.

It comprises genuine videos and photos of the recent past gathered with various kinds of unfavorable climate conditions mainly focusing on dust storms with poor visibility rate and traffic_net dataset. The effectiveness of vehicle recognition is considered as a basic advance in rush hour gridlock observing or smart surveillance. We have also included the calculation and estimation of the type of traffic density being demonstrated in the dataset i.e. dense or sparse, counting the number of vehicles and the forecast of an accident according to the traffic density if it might take place or not and the occurrence of a disastrous fire if in case it happens.

The vehicle detection and counting part are divided into 3 parts- detection, tracking, and counting of course. The detection recognizes the vehicles in the frame of the video and highlights them with a green border around it, and also keeps a track of the vehicle as we don't want to count the same vehicle several times.

The tractor keeps track of the vehicle in different frames of the video. The counter counts the number of vehicles when it crosses the reference line and exits the frame.

To calculate, estimate and predict the net traffic we require Tensorflow, and Keras, OpenCV, and ImageAI installed. Our traffic estimator comprises 4 parts- Accident, Dense Traffic, Fire, Sparse Traffic. For making the program execute we first download_traffic_net() : Downloads traffic net dataset; train_traffic_net() : Train models using imageAI library. This library contains models such as Resnet, Densenet, etc. run_predict() : Function runs prediction

The rest of this paper is organized as follows in section **2**, we have the review of papers. In section **3**, we have the dataset which we have used to train and test our programs. In section **4**, we have listed all the methodologies we have used to dehaze the video and perform all the required operations as mentioned above. In section **5**, we present to you all our results and discussion which we obtained from our code: the picture, the contour images, the confusion matrix MXM matrix having the True label, and the Predicted label. In section **6**, we have the conclusion that we drew from our work our accuracy rate, future scope, and applications, and **7**, references at the last.

II. REVIEW OF PAPER

This paper is focused on traffic generation from video data which is an important application of the intelligent transport system (ITS). It introduces a new tracking system that uses 3D vehicle detection. Vehicle detection and description algorithms are based on a set of magic line functions. , And faster (up to the growth process) and more flexible than previous system algorithms.[1] Tracking after discovery is a common way to track things. As the detection function increases, the reliability of the plotter source increases significantly. This paper focused to demonstrate the ability to use different detectors in a complete experiment. The method presented is easy to apply at 100K fps, although it goes beyond the tracking functions of the DETRAC vehicle.[2] This system proposal includes a new structure that allows multiple vehicles to be reliably identified, classified, and tracked. The system shows excellent performance after an evaluation process with many cameras and various conditions. Two headlights are checked during vehicle detection. If the detector is reliable, the second step looks for clues about the existence of the vehicle via a decision tree. It consists of classifiers according to properties and appearance. The Kalman filter is assigned to the output module. The measuring device is designed for fast, stable, and safe treatment of partial and complete occlusions.[3] This paper is focused on the sand and dust powder (SSS) which are unusual recently on many sides of the world. This is a danger to the environment, healthy economy. There are many techniques to control dust and changes in other environments, such as video-surveillance, watch-out towers, and other research methods. In this, they explain one hybrid method which is based on satellite imaging, wireless devices which will help to predict and detect any type of storm.[4] This paper is focused on car detection and collision avoidance using a machine learning algorithm used for face detection primarily i.e Haar which was first proposed by Viola and Jones. This paper is trying to check if the same method is used for car detection and collision avoidance. What would be its accuracy and efficiency?[5] This paper is focused on pedestrian detection in adverse weather conditions with a method they have proposed: two people detection combining data they received from two sensors a lidar and an ultrawideband radar and then making a fusion of it. Make the model learn and train from it and perform in real-time.[6] This paper is focused on video image processing and detection, tracking, and counting vehicles on road using Raspberry pi3, c++, and OpenCV. The method used is converting RGB to HSV frames and then detecting the vehicles which would help in reducing noise and ease object detection with the color difference, the vehicle tracking is achieved using Kalman filter.[7] This paper puts forward its unique method of vehicle detection for an intelligent vehicle at night time using video and laser information. The method used is first they preprocess the video, then apply Gabor filter and distance of the front vehicle is obtained with the laser point cloud. And the vehicles are classified using the SVM algorithm.[8] This paper presents a traffic monitoring system that is installed on an Unmanned Aerial Vehicle (UAV). This gives results in real-time situations that when tested on a data set give a 4.8% higher precision score and being 91.4% faster when tested on different traffic data. This system runs video at a speed of 29-33fps.[9] This paper focuses on vehicle detection and tracking in adverse weather conditions using deep learning. First, they enhance the visibility which comprises three stages then they detect and track the vehicle using a multi-scale deep convolutional neural network. And have tested their work on 3 primary datasets DAWN, KITTI, MS-COCO, and compared with 21 vehicle detectors.[10] This is a review paper on road traffic accident prediction on high-ranked roads in Europe. The salient features were reviewed and the model was examined in terms of theory, characteristics, data required and the outcomes generated and compared. The results are used in developing APM and CFM inventory which would help other open-source projects.[11] This paper is focused on using IoT for traffic accident detection, the method used is the vehicles interchange their microscopic vehicle variables the data used is collected from VANETs and sends alerts to drivers.

Machine learning algorithms are used such as ANN, SVM, RF to distinguish accidents from normal cases.[12] This paper is using mobile phones to detect accidents and notify the nearest helping agent available for help and immediate action. The paper also states that more accidents happen at low speed than high speed so its first area of interest being the detection of vehicle accidents at low or high speed then notifies and sends information (images, video, coordinates) about the accidents using the smartphone.[13] The paper intends to propose an economic and budget-friendly IoT system that helps is providing help to the passengers stuck in accidents. The application developed would send the details of the accident to the nearest hospital for help.[14] This is a review paper on the different image segmentation algorithms available like segmentation based on clustering, region-based segmentation, segmentation based on weakly-supervised learning in CNN, edge detection segmentation, etc. Then analyze the pros and cons of it and organize them in order of efficiency.[15] This paper presents a new methodology for impurity detection in water using machine vision and has achieved a better recognition rate i.e. 100ms greater diameter and 99% recognition rate.[16] This paper has collected data of adverse weather conditions and formed a data set called ZUT(Zachodniopomorski Uniwersytet Technologiczny) which covers a wide variety of data of European Union countries. And have made it public at Github and IEEE data port.[17] This paper has used a new clustering algorithm and classification is done using SVM. The vehicle detection will be done on cloud point data collected by Velodyne 64 Lidar. Tracking algorithm used is the Kalman filter algorithm and the GNN algorithm.[18] This paper presented a lightweight neural network using Tiny-YOLOv3 and reduced the network which has improved the model accuracy in real-time scenarios and is running at 17 frames per second on a gpu less computer. The main focus of paper is trimming the layers of neural network.[19]

III. DATASET USED

We have used videos available on youtube from the recent past i.e. the sand and dust storms that hit different parts of China in June 2020, the dust/ sandstorms of Arizona, Kuwait, Saudi Arabia, and the Traffic-Net. The Traffic-Net is a dataset containing images of dense traffic, sparse traffic, accidents, and burning vehicles. It contains 4,400 images that span 4 classes. The classes included in this release are:

1 Accident, 2 Dense Traffic, 3 Fire, and 4 Sparse Traffic.

There are 1,100 images for each category, with 900 images for training and 200 images for testing. We are working on adding more categories in the future and will continue to improve the dataset.

Fig. 1. These are the images and frames from Traffic-Net dataset and the videos available online.

IV. METHODOLOGY

The first part of the project is to dehaze the video or image which we are going to use to get accurate and satisfactory results according to our needs. The dehazing consists of 3 parts i.e. image/video manipulation, masking and, the threshold. How we go by this process is at first we input the file either a set of images or a video then we check if its a video or image; if its an image we directly go to the process of image masking and threshold. What is done here is it calculates each pixel of the image according to the matrix we have created and this makes it easy for us to manipulate according to our needs. As we have done here is to dehaze, we can simply say it adds a filter to the image.

Then a threshold value is assigned if the pixel value is greater than the threshold value then an imaginary value is assigned for example 1 if not then 0 is assigned to it, this helps in converting the image to a black and white format for easy modifications on the image. It analyzes the Atmospheric lighting in the image and then gives us a refined and improved image in the output frame.

If a video is given as input then as we know that a video is a collection of several frames played one after the other in a sorted manner. So, what is done here is segregate the video at 25 fps (frames per second) and apply the same process which we have done for the image dehazing, and then all the frames one after the other in a queue and giving us a new refined video as output.

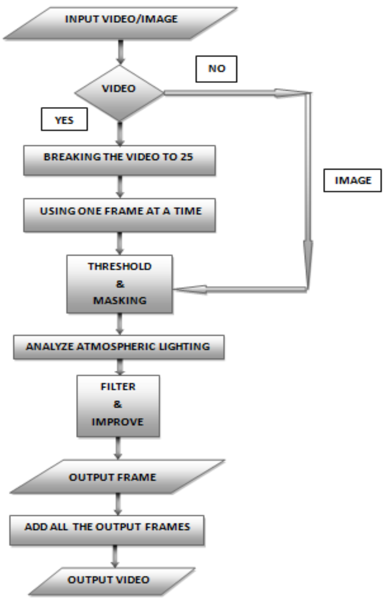


Fig. 2. Flow chart for Image/ Video dehaze.

With the output dehazed video we have, any required operation could be done according to our need. Our target is to detect the vehicle, count the number of vehicles, calculate the traffic density whether it is dense or sparse, predict if an accident or a fire could rage or not.

First, we input the dehazed video which we got as output from the above flow diagram fig 2. Then do preprocess on the video and make it adjustable according to our need and size. We can manipulate it as different videos are of different sizes (aspect ratio) and modes (landscape or portrait).

We apply gaussian blur to the video to reduce the noise and unwanted details in the video to get access to much more accurate results in the outcome. Contour the image which gives us a black and white binary video; these continuous and sharp lines joining all the boundaries help to recognize and identify the moving vehicles in the video. To get a better understanding that the moving vehicle is identified we highlight the vehicle with a neon green boundary around it.

To count the number of vehicles in the video we have used the method of reference line passing objects; a highlighted reference line is brought up on the screen which help helps us to count the number of vehicles that pass from it and a tracker mechanism is used so that the same vehicle is not detected many times in the continuous ongoing frames of the video. The reference line is adjustable as a discrete video has separate camera locations from where the traffic is originating.

To implement the traffic density and prediction of accidents we need to train the model on several photos and videos first to meet our needs to predict and calculate the traffic as it would save the weights for giving accurate values in the future.

After training the model, making it ready for adverse conditions to show its results, that's when the real test of the model comes into play. We also print the confusion matrix and classification report to know the accuracy of the test and understand if more training is needed to be done or not.

Here's a simple flow to make it understandable how the procedure goes on to make it happen.

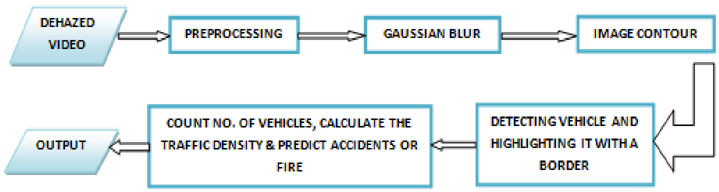


Fig. 3. Flow diagram of vehicle detection, counting, traffic density, forecast of accidents and fire

V. RESULTS AND DICSUSSION

We initially tried to see what the output would be when an unrefined and hazy video was provided to the model, the results were really poor. It couldn't even recognize a single vehicle, in the beginning, the results were like 1 out of 8 vehicles is detected on an average. This proves that if the video is not refined there are severe chances of any unwanted event that could happen because of the lack of visibility. Here are a few results we got at the initial stage.

Fig. 4. Results before training and dehazing the video

As it is quite visible in fig 4 first picture the atmospheric condition is tremendously poor having very little visibility which is causing the problem of not being able to recognize a single vehicle and the contour image so formed is also very poor not able to identify the moving objects which give rise to such outcome. In fig 4 second picture the atmospheric condition is having more brightness and visibility rate as compared to the previous photo but is not crystal clear to identify every vehicle. Here the results are better as compared to the previous one but not the best. It can detect the moving objects and make a good enough binary image using the edges but could be improved, so this gives us a lot of scopes to improve.

Looking at the results it is evident that the videos and images need to be refined so we applied our model of image refinement to get a much clearer and better contrasting image with good lighting, better color, sharper edges which would later help in vehicle detection.

Fig. 5. Results after dehazing.

Here are many results fairly evident that the visibility of the scenes has improved drastically giving much brighter colors, well-defined details, separations between objects, sharp edges, good contrast, and contour. This is done by using a better threshold value in the model we used.

Now, comes the main part of vehicle detection, counting, calculating the traffic density, dense and sparse, calculating and predicting if there are any chances of fire or accidents. Now the results were satisfactory with a great deal of clarity in the images and videos the results were well up to the mark as we expected them to be.

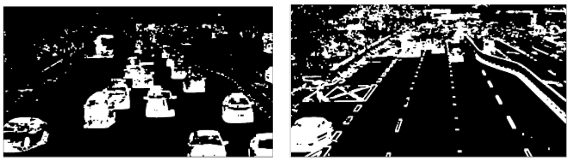


Fig. 6 Contour images while detection of vehicles.

The contour profile has radically improved giving rise to much more vehicles being detected in a frame and providing more accurate results in the calculation of density and prediction of accidents.

Fig. 7 . Results of counting, calculation, and prediction

As we can see in the images the vehicle counting and clarity have increased from the previous time we ran the experiment and it calculates the density of traffic, predicting accidents on the left top of figure 7 above.

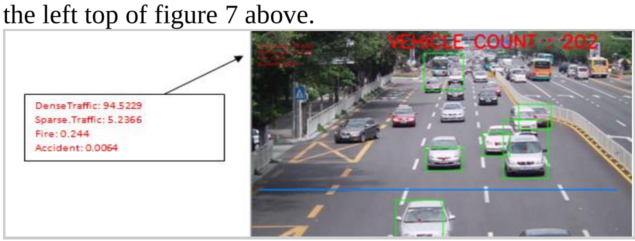


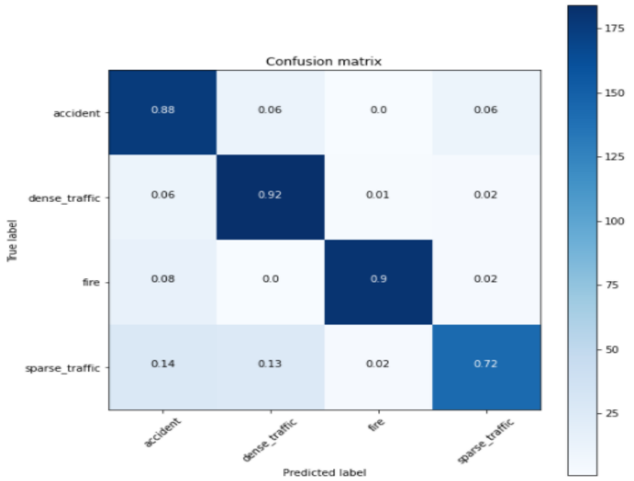
Fig. 8. Calculation and prediction of traffic

Finally, the results of the calculation and prediction show up accurately after training and testing the model on over 3500 images and videos. If the count of vehicles is more than the dense traffic number rises and the sparse traffic number goes down accordingly and vice versa. If the distance between the vehicles is constant or even the accident and fire levels are down if an anomaly is found then the rating of both these goes up enormously defining an uncertain event that might take place.

We have also deduced a confusion matrix and classification table for every data set we use and found out that our accuracy level was good enough to predict the future and save a few lives but the results depend on the data set we give to the model that different dataset would show different accuracy level as no situation is same and the camera angles are different.

VI. CONCLUSION

From the results demonstrated above in fig 5-8, it's quite clear that our motive to achieve a better vehicle detection and estimation of accidents in unfavorable weather conditions have been achieved correctly. Our model to dehaze the data separately and then process the data has provided many efficient results than doing all of it at once which requires much heavier and faster resources to do the computing. The accuracy level it reached is ranging about 85-93% depending on the dataset we provide after the refining is done on the unrefined video.




```
Confusion Matrix
[[176  12   1  11]
 [ 11 184   2   3]
 [ 15   1 181   3]
 [ 28  26   3 143]]
Classification Report
              precision    recall  f1-score   support

           0       0.77      0.88      0.82       200
           1       0.83      0.92      0.87       200
           2       0.97      0.91      0.94       200
           3       0.89      0.71      0.79       200

    accuracy                           0.85       800
   macro avg       0.86      0.85      0.85       800
weighted avg       0.86      0.85      0.85       800
```

Fig. 8. Confusion matrix and classification report.

Here is a confusion matrix that is extracted from testing our system. The result clearly states that it can calculate and estimate the accidents we have shown in the matrix the f1-score which shows the precision levels, recall value, and support.

In this era of advancement self-driving vehicles are going to be the future. The main problem it faces is detecting the roads and lanes in rough weather conditions. That is the motive for writing this paper if this could be an approach to solve a problem for the future. This would not only target a specific area like sand storms or dust storms but any unfortunate weather condition.

For future work, we would like to test the system in real-time, make it self-adjustable to any camera angle, and combine all the different modules into one in an application for the users to make it efficient and easy to use for everyone.

Acknowledgment

We are very grateful to the staff at Computer Science and Engineering lab of Lovely Professional University for their continuous support. We would specially thank our project mentor Ankita Wadhawan for her valuable contribution at various stages of the research.

References

[1] Kim, Z., and Jitendra Malik. "Fast vehicle detection with probabilistic feature grouping and its application to vehicle tracking." In *IEEE International Conference on Computer Vision*, vol. 1, pp. 524-531. 2003.

[2] Bochinski, Erik, Volker Eiselein, and Thomas Sikora. "High-speed tracking-by-detection without using image information." In *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 1-6. IEEE, 2017.

[3] Robert, Kostia. "Night-time traffic surveillance: A robust framework for multi-vehicle detection, classification, and tracking." In *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, pp. 1-6. IEEE, 2009.

[4] Akhlaq, Muhammad, Tarek R. Sheltami, and Hussein T. Mouftah. "A review of techniques and technologies for sand and dust storm detection." *Reviews in Environmental Science and Bio/Technology* 11, no. 3 (2012): 305-322.

[5] Choudhury, Shaif, Soummyo Priyo Chattopadhyay, and Tapan Kumar Hazra. "Vehicle detection and counting using haar feature-based classifier." In *2017 8th annual industrial automation and electromechanical engineering conference (IEMECON)*, pp. 106-109. IEEE, 2017.

[6] Majer, Filip, Zhi Yan, George Broughton, Yassine Ruichek, and Tomáš Krajník. "Learning to see through haze: Radar-based human detection for adverse weather conditions." In 2019 European Conference on Mobile Robots (ECMR), pp. 1-7. IEEE, 2019.

[7] Anandhalli, Mallikarjun, and Vishwanath P. Baligar. "A novel approach in real-time vehicle detection and tracking using Raspberry Pi." *Alexandria engineering journal* 57, no. 3 (2018): 1597-1607.

[8] Zhang, Rong-Hui, Feng You, Fang Chen, and Wen-Qiang He. "Vehicle detection method for intelligent vehicle at night time based on video and laser information." *International journal of pattern recognition and artificial intelligence* 32, no. 04 (2018): 1850009.

[9] Balamuralidhar, Navaneeth, Sofia Tilon, and Francesco Nex. "MultEYE: Monitoring System for Real-Time Vehicle Detection, Tracking and Speed Estimation from UAV Imagery on Edge-Computing Platforms." *Remote Sensing* 13, no. 4 (2021): 573.

[10] Hassaballah, M., Mourad A. Kenk, Khan Muhammad, and Shervin Minaee. "Vehicle detection and tracking in adverse weather using a deep learning framework." *IEEE Transactions on Intelligent Transportation Systems* (2020).

[11] Yannis, George, Anastasios Dragomanovits, Alexandra Laiou, Francesca La Torre, Lorenzo Domenichini, Thomas Richter, Stephan Ruhl, Daniel Graham, and Niovi Karathodorou. "Road traffic accident prediction modelling: a literature review." In *Proceedings of the institution of civil engineers-transport*, vol. 170, no. 5, pp. 245-254. Thomas Telford Ltd, 2017.

[12] Dogru, Nejdet, and Abdulhamit Subasi. "Traffic accident detection using random forest classifier." In *2018 15th learning and technology conference (L&T)*, pp. 40-45. IEEE, 2018.

[13] Ali, Haimd M., and Zainab S. Alwan. Car accident detection and notification system using smartphone. Saarbrucken: LAP LAMBERT Academic Publishing, 2017.

[14] Bhatti, Fizzah, Munam Ali Shah, Carsten Maple, and Saif Ul Islam. "A novel internet of things-enabled accident detection and reporting system for smart city environments." *Sensors* 19, no. 9 (2019): 2071.

[15] Yuheng, Song, and Yan Hao. "Image segmentation algorithms overview." *arXiv preprint arXiv:1707.02051* (2017).

[16] Huang, Bin, Jian Feng Li, Jin Chen Wang, Yin Zhou He, and Wei Shang. "Impurity Detection Using Machine Vision." In *Advanced Materials Research*, vol. 468, pp. 2057-2060. Trans Tech Publications Ltd, 2012.

[17] Tumas, Paulius, Adam Nowosielski, and Arturas Serackis. "Pedestrian detection in severe weather conditions." *IEEE Access* 8 (2020): 62775-62784.

[18] Wang, Heng, and Xiaodong Zhang. "Real-time vehicle detection and tracking using 3D LiDAR." *Asian Journal of Control* (2021).

[19] Tajar, Alireza Taheri, Abbas Ramazani, and Muharram Mansoorizadeh. "A lightweight Tiny-YOLOv3 vehicle detection approach." *Journal of Real-Time Image Processing* (2021): 1-1