◻   184

Measuring the Road Traffic Intensity Using Neural Network with Computer Vision

Muhammad Hamdan[1], Othman Omran Khalifah[2], Teddy Surya Gunawan[3]
[1,3] Department of Electrical and Computer Engineering, Kulliyyah of Engineering
[2] Department of Information Systems, Kulliyyah of ICT
International Islamic University Malaysia

Article Info	ABSTRACT
	Traffic congestion plagues all driver around the world. To solve this problem computer vision can be used as a tool to develop alternative routes and eliminate traffic congestions. In the current generation with increasing number of cameras on the streets and lower cost for Internet of Things (IoT) this solution will have a greater impact on current systems. In this paper, the Macroscopic Urban Traffic model is used using computer vision as its source and traffic intensity monitoring system is implemented. The input of this program is extracted from a traffic surveillance camera and another program running a neural network classification which can classify and distinguish the vehicle type is on the road. The neural network toolbox is trained with positive and negative input to increase accuracy. The accuracy of the program is compared to other related works done and the trends of the traffic intensity from a road is also calculated.

Corresponding Author:

Muhammad Hamdan,
Department of Electrical and Computer Engineering, Kulliyyah of Engineering,
International Islamic University Malaysia,
Jalan Gombak, 53100 Kuala Lumpur, (+603) 6196 4521.
Email: hamdanhg@yahoo.com

1.   INTRODUCTION

Urban development in cities and lowered cost of owning a vehicle has led to rise in number of vehicular in the street causing traffic congestion [1]. Nowadays, it is normal to be stuck in traffic for prolonged period and time is spent for unnecessary part of the day to go through traffic. Figure 1 shows a sample of traffic jam in Asian cities. However, with the current improvement to the availability of internet and vast coverage of the traffic surveillance camera has led many researchers to opt into using computer vision to solve the congestion problem.

In the recent years, advancements in Internet of Things (IoT) as well as lowered cost of owning a smartphone with fast internet connectivity can be used to convey necessary information to the road user. Applications such as Waze and Google map can provide commuters with important details regarding their commute ahead of time and this has helped reducing the traffic jams in cities. However, these applications still depend on the user to send information to the servers to compute the state of the road they are in to convey the information to other. If no users are using the application, then the probability of the congestion occurring is not detected.

Estimation of the amount of traffic on the roads at any given point of time is the first step in mitigating traffic congestion [2]. A common method is to place sensors on the road and count the number of times they are actuated by the passing wheels of a vehicle. This approach suffers from four main problems: a) it is expensive to deploy, as the sensors need to be partially embedded in the tarmac, b) the sensors on the road are prone to theft, c) sensors need to be placed at multiple entry and exit points on the road, to maintain

accurate counts, and d) even on a single stretch of road, the sensors need to be placed at regular intervals to estimate the density on different segments of the road.

Figure 1. Traffic congestion in Kuala Lumpur, Malaysia

Based on the literature review done, the works that shows similarities with current research cope is discussed in detail in this section. In [3], the authors uses blob analysis to track and detect moving cars on the road. Blob analysis uses background subtraction in each frame to generate a blob of moving pixels across the frame. Blob analysis is the simplest way to find moving object in subsequent frames. The author used traffic cameras places in 2 different areas which are cameras facing 3 lanes dubbed as NIPA and TOLL PLAZA with 2 lanes of road. For training the images, Haar Cascade was used to classify the images and XML file was generated using 580 positive images and 1500 negative samples to generate the dataset. Finally, the performance of the system is evaluated using positive detection compared to actual number of vehicles. This paper provided a good platform for this research topic, however the cascade used for this research was based on algorithms used for face tracking and result will be more accurate if the features are more distinct in dataset images.

The next paper focused on using foreground-based method to detect moving object across frames in [4]. Image was acquired using traffic cameras and resized into smaller frames. Then the positive images of cars flowing through the traffic is captured into positive and negative frames. The positive images are then running through the Haar Cascade like the work of [3] but the key difference is the sample was not done using the first frame, but the first number of frames was used to identify the foreground of the images to increase the detection accuracy and removing unwanted noise in each frame being processed. The image with foreground detection becomes more dilated and easier to detect images that are moving. The advantage of this is the amount of processing required to detect the cars are reduced due to smaller ROI but author does not explain on the key performance details such as frame rates or evaluation on the changes in frames sizes on the accuracy.

Another paper discussing on the same issue is written by [5] where the image capturing method was used to identify the number of cars on the road. The procedure used by the writer was background subtraction to identify the objects on the road and measuring them. Instead of measuring the number of blobs, the author counts the number of pixels in which the image is different form the first image with no congestion. By estimation the range of values in these different congestion rates mapped to range of pixel in each state, the conclusion is reached. Key idea of this paper is the simplicity of the algorithm where the images are processed as is and no post processing is required to obtain data however, the data will be not meaningful if the pattern of cars or vehicle types are something important that need to be identified. The author also does not provide any insight on the measurement for the congestion detection systems.

Edge detection is also a key method in image classification tool as researched by [5] which uses IP cameras places in strategic locations on the road to predict traffic congestion. The object detection process is done using edge detection and the parameters are passed onto a fuzzy logic system for traffic estimation. Edge detection in this paper uses a Sobel filter mixed with Kalman filter to track the moving object. The number of moving object is then passed to a fuzzy logic analyzer that considers three parameters such as

vehicle density, distance between vehicle and lastly vehicle size. The logic analyzer then predicts the traffic into different categories based on the logic table of the fuzzer. This method can accurately predict the traffic intensity of the moving traffic however the region of tracking in these images is not set therefore the ROI tracking of moving vehicles must be done on the whole frames which will increase the processing time.

To detect the traffic intensity, there are 3 critical approach in the modelling of the traffic flows in roads namely the spatial modelling [6]–[9], time prediction modelling [10]–[12] and non-parametric modelling. Spatial modelling refers to using the physical space on real time to estimate the motion of the vehicles based on the physical parameters such as road distance, car length, road curvature and other key elements. Time prediction models use statistical data collected over time to predict the current trend on the road and update the data on real time if necessary. Non-parametric modelling refers to using computational intensive algorithms such as artificial neural network and work independent from any real-time input but relies on collection of data to synthesize an output. However, in this implementation the space based modelling technique is used to simplify the result.

2.   RESEARCH METHOD

Figure 2 shows the overall implementation of the proposed solution. The process starts when the traffic camera is trained on the foreground and background images on an area. Then the program displays the images to the user to identify the Region of Interest(ROI) for the desired location to start counting the cars. The user will choose the lower and upper region of the ROI to determine traffic flow into and out of the link as shown in following Figure 3.
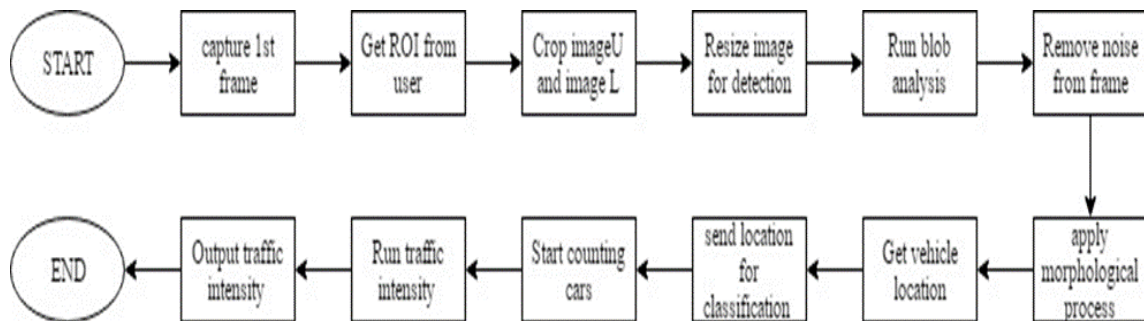

Figure 2 The overall process to identify traffic intensity

The figure shows the overall implementation of the proposed solution. The process starts when the traffic camera is trained on the foreground and background images on an area. Then the program displays the images to the user to identify the Region of Interest (ROI) for the desired location to start counting the cars. The user will choose the lower and upper region of the ROI to determine traffic flow into and out of the link as well as camera placement is shown in following Figure 3. This method is not implemented in other research done in the literature review as most works runs tracking on entire video frame.
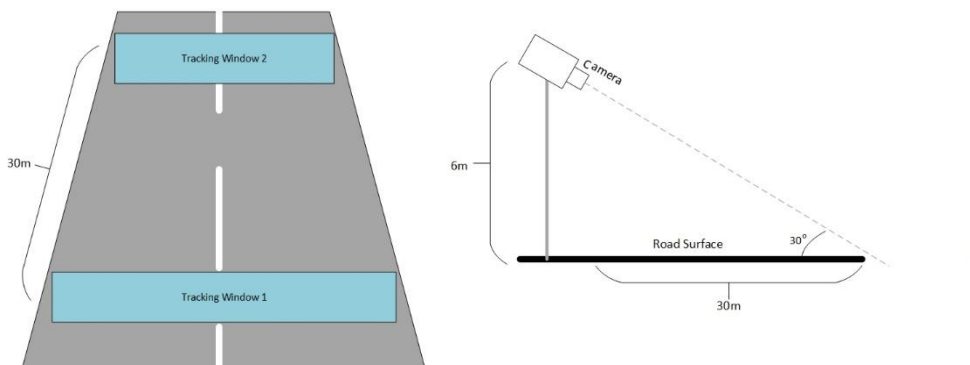

Figure 3. Tracking location of vehicles for traffic intensity and camera placement

Next the frame captured from camera subtracted from the current frames and the differences are marked using blob analysis and its output is filtered for regions that are larger than 5000 pixels which is dependent on the original video size. Next the centroid of the moving objects is calculated from the blob and it is passed to the classification tool.
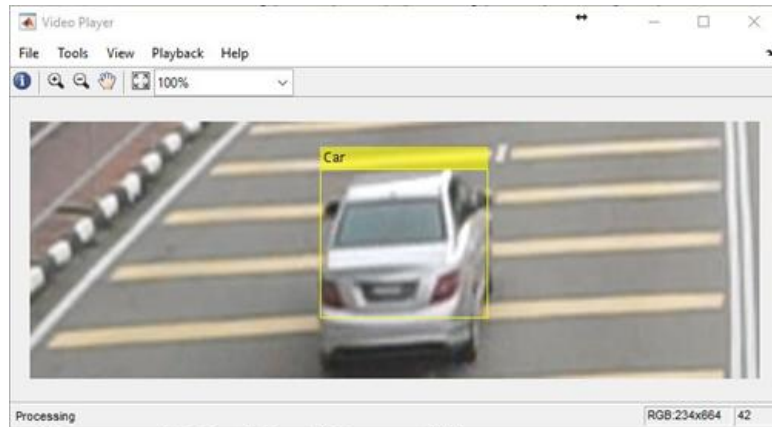

Figure 4. GUI of the car using Matlab

The classification tool contains the pretrained object output of images from vehicles given before the program is initialized. The training process involves using positive images of cars and other vehicles and negative image containing background and unrelated random images to allow the computer to differentiate the type of object given to it. The object file is generated using neural network toolbox in Matlab which gives a high accuracy of data for object classification. When the image is passed to this program, it will determine if the centroid given to it contains which vehicle from the trained object.

3.   RESULTS AND DISCUSSION
3.1. Program Accuracy
The program is run on a road and the data collected is discussed in this section. The sample location selected was in Penang, Malaysia where a camera is placed on the center of the road facing the link. The accuracy of the classification program is as shown in Table 2. This result shows the optimum condition is used for data collection and image processing. The assumptions used for vehicle tracking is the car is moving in a constant velocity and speed below 60km/h as faster vehicles tracking is not practical to track using background subtraction.

Table 1. Overall Accuracy

Type of vehicle detected	Number of vehicle detected	Number of vehicle undetected	Accuracy (%)	Average Detection time(ms)
Car	85	15	85	150
Motorcycle	30	7	80	150

The program can tell with almost 85 percent probability if the detected object is a car and 80 % accuracy of the detected object is a motorcycle. Based on these data, it provides a good reference to optimize the system in the future though is slightly lower than [3],[6]. Next the program is run during the night time to get the accuracy when the condition is dark and Table 3 shows the accuracy of the data collected on the same location.

Table 2. Accuracy of the program during night

Type of vehicle detected	Number of vehicle detected	Number of vehicle not detected	Accuracy (%)	Average Detection time (ms)
Car	8	24	25	220
Motorcycle	2	15	11	230

The data in Table 3 shows that the accuracy has dropped significantly compared to the video captured during the day. These results are not compared to other researches as this test was not done by them. There are a few factors that contributed to this drop in the accuracy such as different lighting on the road causing the changes in the shade of car be lower than it should be when the blob analysis code is running. Next, the car at night uses light to illuminate the road thus changing a key feature extracted during training process. The light projected by the car also increased brightness that makes the camera unable to see the actual car shape but only see circle during blob analysis. Lastly, the colors of the light color also change when the car is moving forward and backward as the headlight is white and the break light is red.

3.1. Traffic Intensity

The traffic intensity during low traffic and high traffic movement is calculated and the result is as shown in Table 3 and 4.

Table 3. Traffic intensity during low congestion

Time Start	Time end	Number of cars into link(per minute)	Number of car out of link(per minute)	Traffic intensity at time k (no Congestion)
0	1	26	25	0.018367347
1	2	9	10	-0.017307692
2	3	7	6	0.017045455
3	4	3	4	-0.016917293
4	5	22	21	0.018072289
5	6	12	12	0
6	7	4	5	-0.016981132
7	8	11	10	0.017307692
8	9	13	13	0
	Total	107	106	0.019586665

Table 4. Traffic intensity during medium to high congestion

Time Start	Time end	Number of cars into link(per minute)	Number of car out of link(per minute)	Traffic intensity at time k (Congestion present)
0	1	20	17	0.053359684
1	2	17	14	0.052734375
2	3	22	20	0.036
3	4	19	17	0.035573123
4	5	20	21	-0.018072289
5	6	22	20	0.036
6	7	13	13	0
7	8	23	21	0.036144578
8	9	20	23	-0.05465587
	total	176	166	0.1770836

When there is no congestion, the total traffic intensity in each time is low and the traffic intensity during peak hours are high therefore the program will be able to tell the difference in the traffic intensity clearly. The user can set the threshold where the traffic intensity is high, medium or low depending on the road condition as explained in the research paper [13].

Known limitation of the implementation are shadows, car color and performance. During light condition changes, the accuracy of the program varies due to shape of shadow increases and decreases over time. Another limitation is the car colors like black and grey causes the system to not detect the moving object between frames as it is the same color of the road hence reducing the accuracy. Finally, the performance drops significantly if more than 4 vehicles are tracked at the same time however the frames size used typically contains about 3 cars maximum. To better evaluate the performance of our proposed method, a traffic crowd simulation based on intelligent agent could be developed.
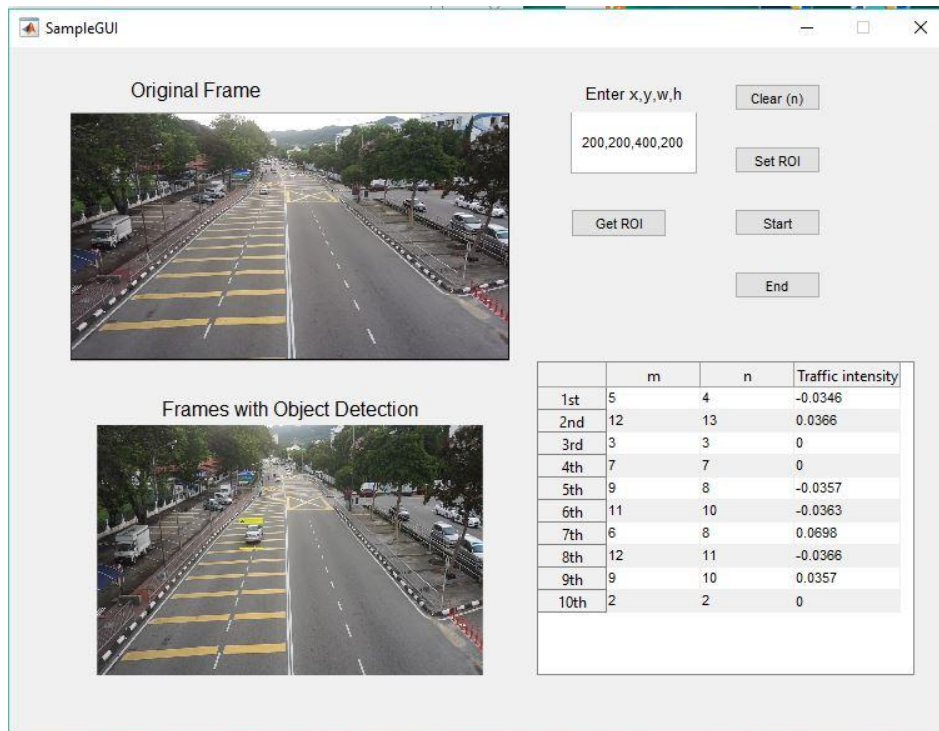
Figure 5. Sample GUI Created to Display Output

4. CONCLUSION AND FUTURE WORKS

In this paper, a method of identifying the traffic intensity of the road is implemented using computer vision. Based on the data acquired, the program can tell accurately if the road is congested based on the equation of the Macroscopic Urban Traffic models. However, using computer vision still has limitations as discussed in the previous section. In the future, a better program which collect multiple road statistics and generate a prediction algorithm based on the road trends is needed to be able to tell road users information about their journey on the road. Moreover, a traffic crowd simulation could be developed to evaluate our method.

REFERENCES

[1] A. S. Abdelfatah, *et al.*, "Evaluating the Sustainability of Traffic Growth in Malaysia," *J. Traffic Logist. Eng.*, vol/issue: 3(1), pp. 6–11, 2015.
[2] A. Asahara, *et al.*, "Macroscopic Structural Summarization of Road Networks for Mobile Traffic Information Services," in *7th International Conference on Mobile Data Management (MDM'06)*, pp. 42–42, 2006.
[3] Y. Hasan, *et al.*, "Comparative analysis of vehicle detection in urban traffic environment using Haar cascaded classifiers and blob statistics," *FTC 2016 - Proc. Futur. Technol. Conf.*, pp. 547–552, 2017.
[4] Q. Wu, *et al.*, "Real-time vehicle detection with foreground-based cascade classifier," *IET Image Process.*, vol/issue: 10(4), pp. 289–296, 2016.
[5] B. Eamthanakul, *et al.*, "The Traffic Congestion Investigating System by Image Processing from CCTV Camera," in *2017 International Conference on Digital Arts, Media and Technology (ICDAMT)*, pp. 240–245, 2017.
[6] F. S. Fernandez, *et al.*, "Background subtraction for aerial surveillance conditions," *14th Int. Conf. Innov. Community Serv. "Technologies Everyone", I4CS 2014 - Conf. Proc.*, pp. 28–33, 2014.
[7] K. Phalke and R. Hegadi, "Object tracking using Mutiple Hypothesis," in *2016 3rd International Conference on Signal Processing and Integrated Networks (SPIN)*, pp. 707–710, 2016.
[8] F. Yan, *et al.*, "An improved traffic network model for model-based predictive control," in *Proceedings of the 33rd Chinese Control Conference*, pp. 3405–3410, 2014.

[9]   A. Boulmakoul, *et al.*, "Towards Scalable Distributed Framework for Urban Congestion Traffic Patterns Warehousing," *Appl. Comput. Intell. Soft Comput.*, vol. 2015, 2015.

[10] L. Li, "A continuum model of traffic flow on road networks," in *2015 International Conference on Transportation Information and Safety (ICTIS)*, vol. 2, pp. 595–598, 2015.

[11] L. Sindhia, *et al.*, "Efficient detection tracking of multiple moving objects in temporal domain," in *2016 International Conference on Recent Trends in Information Technology (ICRTIT)*, pp. 1–6, 2016.

[12] A. Ouessai and M. Keche, "Development of road traffic flow model for nighttime conditions," *3rd Int. Conf. Control. Eng. Inf. Technol. CEIT 2015*, 2015.

[13] H. G. M. Hamdan, *et al.*, "Traffic intensity monitoring using multiple object detection with traffic surveillance cameras," *IOP Conf. Ser. Mater. Sci. Eng.*, vol/issue: 260(1), pp. 28–30, 2017.