An Adaptive Video-based Vehicle Detection, Classification, Counting, and Speed-measurement System for Real-time Traffic Data Collection

Amit Ghosh, Md. Shahinuzzaman Sabuj, Hamudi Hasan Sonet,
Swakkhar Shatabda, and Dewan Md. Farid*
Department of Computer Science & Engineering, United International University
United City, Madani Avenue, Badda, Dhaka 1212, Bangladesh
Email: amitbd1508@gmail.com, msabuj132023@bscse.uiu.ac.bd, hhsonet@gmail.com
swakkhar@cse.uiu.ac.bd, dewanfarid@cse.uiu.ac.bd

***Abstract*—Intelligent Transportation System (ITS) is an integral part for efficiently and effectively managing road-transport network in metros and smart cities. ITS provides several important features including public transportation management, route information, safety and vehicle control, electronic timetable and payment system etc. In this paper, we have designed and developed an adaptive video-based vehicle detection, classification, counting, and speed-measurement tool using Java programming language and OpenCV for real-time traffic data collection. It can be used for traffic flow monitoring, planning, and controlling to manage transport network as part of implementing intelligent transport management system in smart cities. The proposed system can detect, classify, count, and measure the speed of vehicles that pass through on a particular road. It can extract traffic data in csv/xml format from real-time video and recorded video, and then send the data to the central data-server. The proposed system extracts image frames from video and apply a filter based on the user-defined threshold value. We have applied MOG2 background subtraction algorithm for subtracting background from the object, which separates foreground objects from the background in a sequence of image frames. The proposed system can detect, classify, and count vehicles of different types and size as a plug & play system. We have tested the proposed system at six locations under different traffic and environmental conditions in Dhaka city, which is the capital of Bangladesh. The overall average accuracy is above 80% for classifying all types of vehicles in Dhaka city.**

***Keywords*—Intelligent Transportation System; Traffic Data Collection; Smart City; Vehicle Classification;**

I. INTRODUCTION

A smart city is a framework that aims at increasing the quality of urban life through improved decision-making using information and communication technologies. An Intelligent Transport System (ITS) is an essential facet of an advanced smart city system. The purpose of ITS improve safety, mobility and efficiency of the traffic system by processing information to minimise congestion, environmental impact and sustain the benefits of transportation [1], [2], [3]. ITS provides the users with information about optimised travel routes, real-time traffic data and predictions, passenger density and seat availability of vehicles and many more [4], [5], [6]. The working model of an ITS can be divided into stages of data collection, transmission, analysis and relay to users [7]. Precise and prompt real-time data of traffic count, location, vehicle weight, delays etc. are collected through hardware devices such as sensors, cameras, and GPS locators and transmitted over the internet to the server for further analysis [8], [9]. The collected data is screened, corrected, synthesised and pooled and later analysed to predict traffic situations accordingly. Information regarding travel time, route, congestion, accidents, roadblocks etc. is then extracted from the analysis and sent to the users. With the available information, users are able to make more efficient, safer and smarter use of the transportation network to suit their needs. ITS models are emerging globally, and the opportunities for solving modern urbanisation challenges are also growing in this digital era.

Computer vision technology is using for traffic monitoring in many countries [10], [11]. The development of computer vision technology over video based traffic monitoring for detecting moving vehicles in video streams become an essential part in ITS [12], [13]. A good number of work has been done on vehicle tracking and detection using computer vision technology. In 2005, Hasegawa and Kanade [14] introduced a system for detecting and classifying the moving objects by its type and colour. In this process, a series of images of a specific location were supplied and vehicles from these images were identified. In 2013, Nilesh et al. [15] designed and developed a system using visual C++ with OpenCV for detecting and counting moving vehicles. It can automatically identify and count moving objects as vehicle in real-time or from recorded videos, which basically used background subtraction, image filtering, image binary and segmentation method. In 2014, Da Li et al. [16] developed real-time moving vehicle detection, tracking, and counting system also using Visual C++ with OpenCV including adaptive subtracted background method in combination with virtual detector and blob tracking technology. Virtual detector constructs a set of rectangular regions in each input image frame and blob-tracking method generates input image frames, the absolute difference between the background image and foreground blobs corresponding to the vehicles on the road. The above systems have some limitations like tackling shadows, occlusion of multiple vehicles that appear in a single region. Peek Traffic Corporation (https://www.peektraffic.com/index.php) commercially developed several video traffic detection systems at the present time.

978-1-7281-0297-9/19/$31.00 ©2019 IEEE

In this paper, we have proposed an adaptive video-based vehicle detection, classification, counting, and speed-measurement system for real-time traffic data collection. The proposed system was built using Java programming language (https://www.java.com/) and OpenCV (https://opencv.org). The main objective for developing this system is to collect vehicle count and classification data. So that we can build intelligent transportation network based on historical traffic data. The proposed system can engender traffic data by detecting, classifying, counting, and measuring speed of vehicles and store the in csv/xml file format. It's a plug & play system and applied MOG2 algorithm as a background subtraction technique. The proposed system was tested at different six locations in Dhaka city (Dhaka is the capital of Bangladesh) under different traffic and environmental conditions. It achieved average 81% accuracy for classifying all types of vehicles in Dhaka city.

The rest of this paper is organised as follows. Section II presents the methodology that we have used for building the proposed system. Section III presents the graphical user interface (GUI) of the proposed system. Section IV provides experimental test results. Finally, Section V presents conclusions and future work.

II. Methodology

A. System Overview

The proposed system can run in two modes: recorded video mode and in real-time mode. It converts video into a sequence of image frames, and then extracts background and performs detection of moving objects. Video source may be of two types: (1) recorded video, and (2) real-time video. Initially, an area-threshold is set for vehicle detection. A count line and speed line is drawn to count and measure the speed of detected vehicles. Fig. 1 and Fig. 2 show the flow chart and activity diagram of the proposed system respectively. Several stages of the proposed system are described in details in the following sub-sections.

B. System Initialisation

The proposed system is get installed and set up in this stage. In this stage, we setup cameras on lamppost or pillar of roads in an angle so that we can have the clear view of the road. Count line and speed line is drawn and we also set the distance threshold between the speed line and the count line which is the real distance in meter. We set up other thresholds like minimum size of vehicle, image threshold etc. Camera records stream of data from roads and sends to the system for further analysis.

C. Image Digitisation and Background Subtraction

Background subtraction is a broadly applied method for detecting moving objects in video from static cameras. The proposed system is used background subtraction technique for object extraction from video. It's faster, very dynamic and produces a better result than other pattern-based [17] detection. Pattern-based detection is good for detecting unique objects. But, when there is a variety in types of vehicles, pattern-based detection can't produce a better result. In background subtraction technique the moving foreground is extracted from a static background. Background subtraction performs a subtraction between current frame and the background model to determine the foreground mask. Fig. 3 shows only the moving objects after performing background subtraction from a recorded video.

D. Vehicle Detection & Classification

There are only the moving vehicles after performing background subtraction. The system detects each moving vehicle and the detected vehicle is surrounded with a rectangle. The size of the rectangle refers to the area of the detected vehicle. In the proposed system a minimum area can be setup as area-threshold. If the size of a moving object is greater than the defined minimum area-threshold, it will only be considered as a vehicle then, otherwise it will be ignored. That is why human will not be detected in the system. Fig. 4 shows the detected vehicles. To classify the detected vehicles in different types, like large vehicles, medium vehicles and small vehicles, we again consider the size. The size range has been defined earlier for the different types of vehicles in the proposed system. When a detected vehicle size lies in a particular size-range, the vehicle is classified in that specific vehicle type.

E. Vehicle Counting & Speed Measurement

While counting vehicles, it's very important to count each vehicle only once. For this purpose, a count line has been introduced in the system. When detected vehicles pass over the count line will only be counted otherwise not. The proposed system also has a speed line to measure the speed of moving vehicles. While configuring the system, it is needed to draw both count line and the speed line. The distance between the count line and the speed line will be fixed earlier at the time of setting up the camera. As the distance between the speed line and the count line is known and our developed system can calculate the time for crossing the speed line to count line. Thus the system can easily calculate the speed of a vehicle.

F. Shadow Identification and Removal

If there is a shadow of a vehicle and the shadow is moving as well, the background subtraction considers it as moving foreground and thus the shadow may be detected as a moving vehicle. As simple background subtraction can't detect the moving shadow of vehicles, some good algorithms are introduced like BackgroundSubtractorMOG2 etc. to deal with the shadow problem. In BackgroundSubtractorMOG2 [18] algorithm, there has an option for selecting whether a shadow is to be detected or not. When detectShadows = True, the shadows are detected and BackgroundSubtractorMOG2 marks shadows and shadows are marked in gray color. For shadow identification and removal, it has been used the BackgroundSubtractorMOG2 in the proposed system.

III. Graphical User Interface

The functions and functionalities of the proposed system are shown in Fig 5 and described below:

1) 1.1 Represents the rectangle, which defines the area.

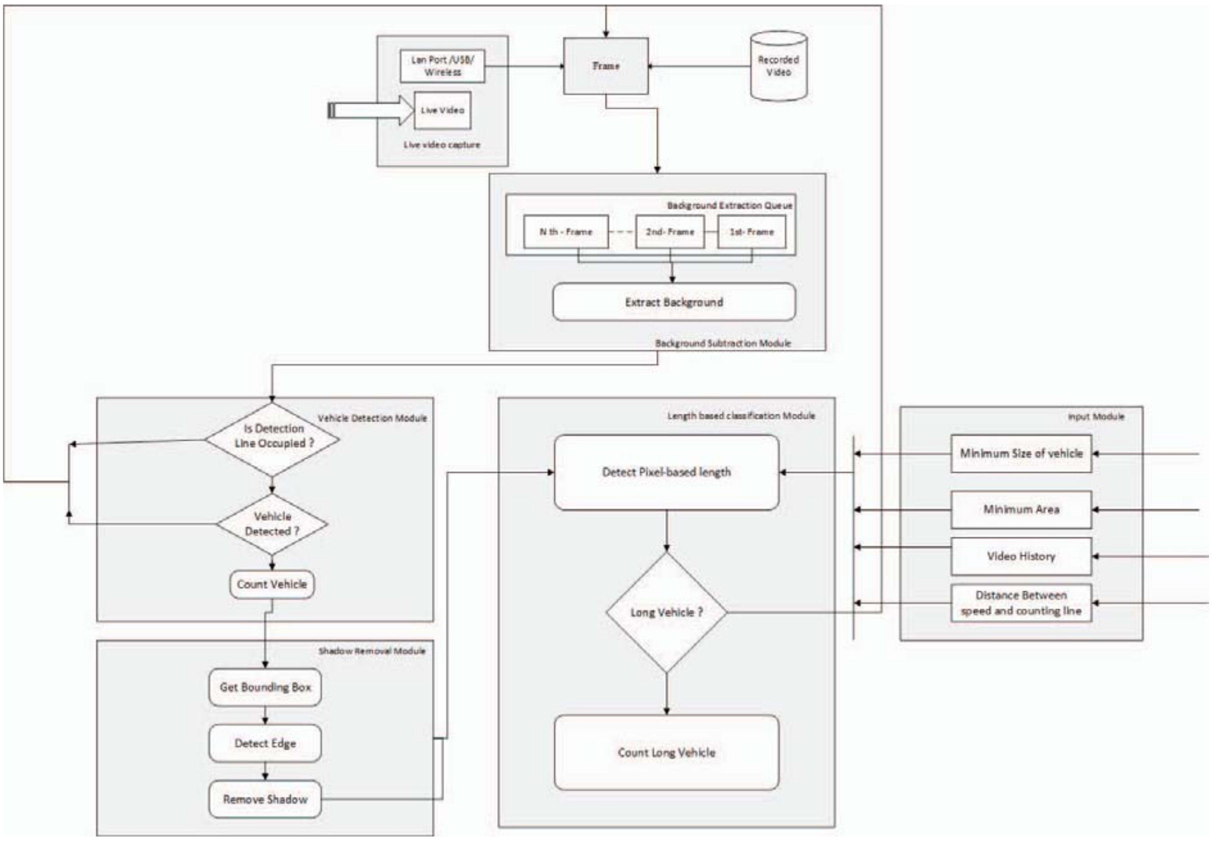


Fig. 1: Flow chart of the proposed system.

- 1.2 Shows the area of the corresponding rectangle that is actually the detected vehicle size.
- 1.3 The red line appears is the count line. All detected vehicles pass through this line will be counted.
- 1.4 The speed line which is marked as green is measuring the speed of the passing vehicles.
- 1.5 The car enclosed by the rectangle is the detected object from the video.

2) Shows the subtracted background of objects in real time.
3) This input field indicates the location of the recorded video.
4) Although the data-set name is automated, user can set a default name of a specific data-set.
5) Shows the button to draw the count line.
6) Noticed the button to draw the custom speed line. The line needs to cover the whole road.
7) Indicates the path to save the data-set captured via camera which we can also chose manually .
8) For loading the recorded video in the system.
9) It is to select the camera which is attached to the device.
10) Resets all the changes to initial state.
11) Plays all the process and pauses all the process.
12) Stops all the process and save the data-set in the chosen path.
13) To define the minimum size of vehicles. The value can be changed at any time according to the camera position. We have considered this threshold size to deduct the walking people in the road, tiny creatures like birds that may also appear in front of the camera.
14) To define the minimum size of objects that to be detected.
15) How much time an object will remain in the BGS view after it first appears.
16) This is a threshold value for controlling the darkness of an image. When an image is subtracted from a known background the colour, contrast, brightness is a very important issue. So we need to use this threshold for adjusting the colour in a different situation.
17) The path for saving video where all the recorded video will be stored. It saves current-time and date and also a determine the extension name.
18) Save file in csv or xls format.
19) The actual distance between the count line and speed line. Speed calculation depends on this threshold.

IV. Experimental Tests

We have tested the proposed system on a laptop powered by an Intel Core i5-3230M processor (2.6 GHZ) CPU and 8 GB RAM equipped with 720p HD web-cam on image sequences of city roads. The system is able to track and classify most

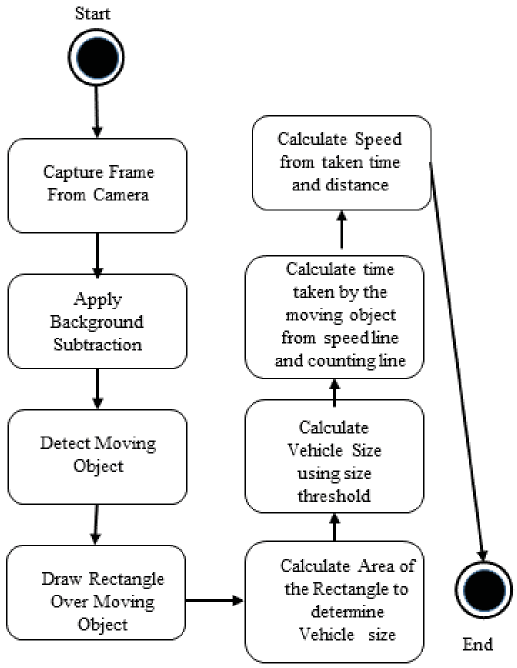


Fig. 2: Activity diagram of the proposed system.

vehicles successfully. We have tested the proposed system with real-time and recorded video at six different locations under different traffic and environmental conditions in Dhaka city, which is the capital of Bangladesh. Test results are tabulated in Table I and Table II.

TABLE I: Result on real-time video

Location	Data	Minute	Accuracy
Mirpur Road	1090	45	70 %
Satmosjid Road	510	15	87 %
Dhanmondi 27	460	30	89 %
Total	2320	90	82 %

TABLE II: Result on recorded video

Location	Data	Minute	Accuracy
Farmgate	380	20	70 %
Gulsan 2	690	20	87 %
Dhanmondi 15	460	25	85 %
Total	1530	65	81 %

Table III shows the feature details of dataset that generated by the proposed system. Table IV shows some sample instances that generated by the proposed system.

V. Conclusions & Future Work

The demands of Intelligent Transport System (ITS) are increasing gradually and steadily. We have developed video-based vehicle detection, classification, counting, and speed-measurement adaptive system for real-time traffic data collection. We have used BackgroundSubtractionMoG2 algorithm, OpenCV, and Java SE Development Kit 8 for developing the

TABLE III: Attribute description of generated dataset.

Attribute Name	Type	Description
No.	numeric	Number of vehicles detected
Vehicle type	nominal	Different types of vehicles
Speed [km/h]	numeric	Speed of vehicles
Size	numeric	Different vehicles size
Date	yyyy/mm/dd	Date of data collection
Time	hh:mm:ss	Time of data collection

TABLE IV: Sample traffic dataset generated by the proposed system.

Sl.	Vehicle Type	Speed [Km/h]	Size[$pixel^2$]	Date	Time
1	BUS	28.4818212	11278	2017/10/30	15:51:52
2	CNG	38.618212	2941.5	2017/10/30	15:52:02
3	BIKE	25.64134813	1823	2017/10/30	15:52:05
4	CAR	31.84826663	4158	2017/10/30	15:52:07
5	BIKE	46.73896764	1883.5	2017/10/30	15:52:10
6	BIKE	53.20573051	1789.5	2017/10/30	15:52:10
7	CNG	27.17428227	2887	2017/10/30	15:52:11
8	CAR	34.76447177	6135	2017/10/30	15:52:18
9	CAR	60.09590226	7969.5	2017/10/30	15:52:23
10	CNG	54.68656872	2444	2017/10/30	15:52:25
11	BIKE	48.71949885	1895.5	2017/10/30	15:52:25
12	BIKE	52.64005541	1728	2017/10/30	15:52:25
13	CNG	41.53079131	2792.5	2017/10/30	15:52:27
14	CAR	46.39157098	3727	2017/10/30	15:52:28

system. We have used git for remote repository. In the proposed system, we have considered all day and night shadowing, and different lighting situations. Also, we have considered the moving shadow of moving vehicles. We have tested the proposed system at six different locations under different traffic and environmental conditions in Dhaka city. The proposed system can generates traffic data by analyzing video with classification of vehicle type, vehicle length, vehicle speed, and time & date. It can generate traffic data in csv or xml file format. It's a plug & play system. The experimental results indicates that the proposed system can effectively detect, classify, count, and measure speed of moving vehicles with above 80% accuracy. In future, we will collect more traffic data from different locations of Dhaka city using our proposed system and will apply several machine learning algorithms for mining traffic patterns of Dhaka city.

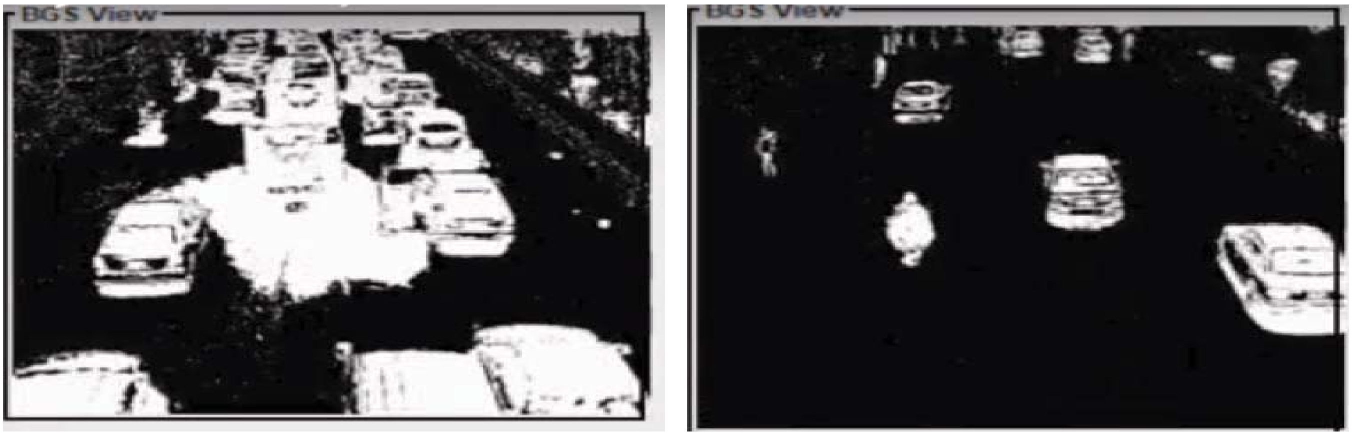
Acknowledgment

We would like to thank "a2i (Access to Information) Innovation Lab" (https://a2i.gov.bd/innovation-lab/) for considering this project, which is the winner of "National Innov-A-thon 2018" held by the ICT division (https://ictd.gov.bd) of the Government of the People's Republic of Bangladesh.

References

[1] I. Alam, M. F. Ahmed, M. Alam, J. Ulisses, D. M. Farid, S. Shatabda, and R. J. F. Rossetti, "Pattern mining from historical traffic big data," in *IEEE Technologies for Smart Cities (TENSYMP 2017)*. IEEE, July 2017, pp. 1–5.

[2] I. Alam, D. M. Farid, and R. J. F. Rossetti, "The prediction of traffic flow with regression analysis," in *Emerging Technologies in Data Mining and Information Security*, ser. Advances in Intelligent Systems and Computing, A. A., D. P., M. J., B. A., and D. S., Eds. Springer, Singapore, 2019, vol. 813, pp. 661–671.



(a) (b)

Fig. 3: Background Subtraction: (a) Initial state; (b) Stable state.

(a) (b)

Fig. 4: Vehicle Detection: (a) Drawing rectangle; (b) Counting vehicle.

[3] A. Talebpour, H. S. Mahmassani, and S. H. Hamdar, “Effect of information availability on stability of traffic flow: Percolation theory approach,” *Transportation Research Procedia*, vol. 23, pp. 81–100, 2017.

[4] M. Dell’Orco and M. Marinelli, “Modeling the dynamic effect of information on drivers’ choice behavior in the context of an advanced traveler information system,” *Transportation Research Part C: Emerging Technologies*, vol. 85, pp. 168–183, 2017.

[5] A. Csikós, T. Charalambous, H. Farhadi, B. Kulcsár, and H. Wymeersch, “Network traffic flow optimization under performance constraints,” *Transportation Research Part C: Emerging Technologies*, vol. 83, pp. 120–133, 2017.

[6] M. Zhou, X. Qu, and X. Li, “A recurrent neural network based microscopic car following model to predict traffic oscillation,” *Transportation Research Part C: Emerging Technologies*, vol. 84, pp. 245–264, 2017.

[7] F. B. Ghorghi and H. Zhou, “Traffic control devices for deterring wrong-way driving: historical evolution and current practice,” *Journal of Traffic and Transportation Engineering*, vol. 4, pp. 280–289, 2017.

[8] A. Abadi, T. Rajabioun, and P. A. Ioannou, “Traffic flow prediction for road transportation networks with limited traffic data,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 653–662, 2015.

[9] X. Chen, L. Peng, M. Zhang, and W. Li, “A public traffic demand forecast method based on computational experiments,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 4, pp. 984–995, 2017.

[10] S. Gupte, O. Masoud, R. Martin, and N. Papanikolopoulos, “Detection and classification of vehicles,” *IEEE Transactions on intelligent transportation systems*, vol. 3, no. 1, pp. 37–47, August 2002.

[11] J. J. Kim, S. Smorodinsky, M. Lipsett, B. C. Singer, A. T. Hodgson, and B. Ostro, “Traffic-related air pollution near busy roads: the east bay children’s respiratory health study,” *American journal of respiratory and critical care medicine*, vol. 170, no. 5, pp. 520–526, May 2004.

[12] A. H. Lai, G. S. Fung, and N. H. Yung, “Vehicle type classification from visual-based dimension estimation,” *Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE*, pp. 201–206, August 2001.

[13] A. Rhodes, D. Bullock, J. Sturdevant, Z. Clark, and D. C. Jr, “Evaluation of the accuracy of stop bar video vehicle detection at signalized intersections,” *Transportation Research Record: Journal of the Transportation Research Board*, no. 1925, pp. 134–145, 2005.

[14] O. Hasegawa and T. Kanade, “Type classification, color estimation, and specific target detection of moving targets on public streets,” *Machine Vision and Applications*, vol. 16, no. 2, pp. 116–121, December 2005.

[15] N. J. Uke and R. C. Thool, “Moving vehicle detection for measuring traffic count using OpenCV,” *Journal of Automation and Control*

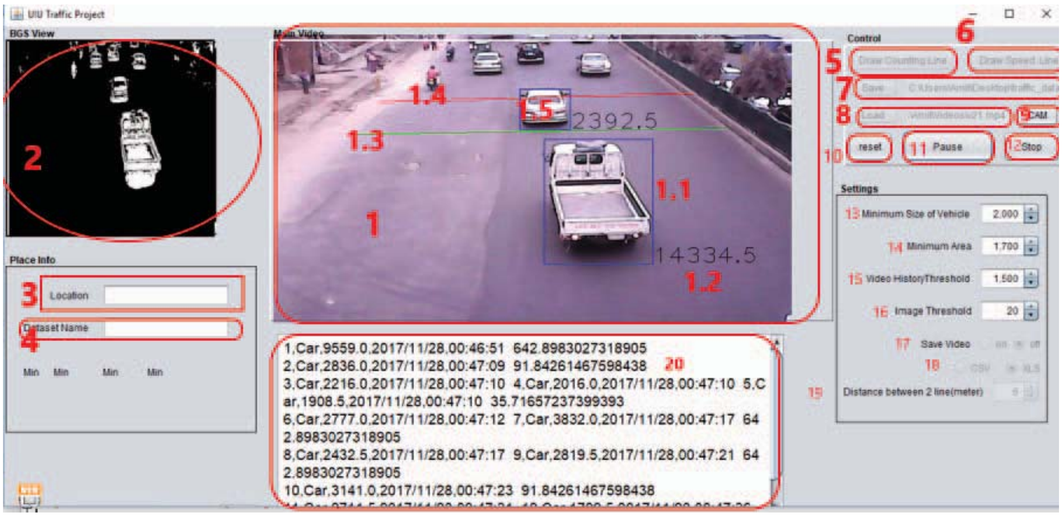


Fig. 5: GUI of the proposed system.

Engineering, vol. 1, no. 4, pp. 349–352, December 2013.

[16] D. Li, B. Liang, and W. Zhang, "Real-time moving vehicle detection, tracking, and counting system implemented with OpenCV," *2014 4th IEEE International Conference on Information Science and Technology*, pp. 631–634, October 2014.

[17] S. Deep, R. A. Shaikh, J.-P. Li, M. H. Memon, A. Khan, and Z. Tao, "Pattern based object recognition in image processing," *2014 11th International Conference on Wavelet Activ Media Technology and Information Processing(ICCWAMTIP)*, pp. 276–279, December 2014.

[18] P. KaewTraKulPong and R. Bowden, "An improved adaptive background mixture model for real-time tracking with shadow detection," *Video-based surveillance systems*, pp. 135–144, 2002.