Intelligent Vehicle Counting for Video Surveillance Application

Maria Crystal E. Orozco[1], Corazon B. Rebong[2]
[1,2] Technological Institute of the Philippines, Philippines
[1]orozco.mariacrystal@gmail.com, [2]corazonbrebong@gmail.com

ABSTRACT

In most developed cities in the Philippines, video surveillance systems are presently implemented, which results in vast amounts of video data stored in servers. In a particular city, most CCTV cameras capture traffic scenes in urban roads. Moreover, command centers and traffic management office need this information to respond to emergencies and get visual information on traffic situations. CCTV operators monitor these live-feed data to record and report any incidents that have happened 24/7. With this, little or no time is given for video analysis, such as counting vehicles, which is a piece of essential information in planning and making decisions for the betterment of urban traffic flow and readiness to emergency events. It is for this reason that the researchers proposed an intelligent vehicle counting to be integrated into video surveillance applications using Tensorflow object detection and counting APIs, a deep-learning approach to classify and count vehicles. Initially, a classifier has been built and trained to detect and count vehicles and distinguish them from other objects in the sample captured videos. The newly-trained vehicle classifier is tested and evaluated using recall, precision, and F1-score performance metrics.
On the other hand, the vehicle counter has been run and tested for different positions of the region of interest (ROI) line and eventually evaluated using the accuracy metrics. Results show that the said system is recommended to be integrated into video surveillance systems. However, a fraction of detection and counting errors is noted for future enhancements of the system.

Keywords
deep learning, Surveillance System, Tensorflow, vehicle classification, detection and counting

Article Received: 10 August 2020, Revised: 25 October 2020, Accepted: 18 November 2020

Introduction

Traffic congestion has been a significantly challenging problem. It has widely realized that increases of preliminary transportation infrastructure, more pavements, and widened road, have not been able to relieve city congestion. As a result, many investigators paid their attention to intelligent transportation systems (ITS), to predict the traffic flow based on monitoring the activities at traffic intersections for detecting congestions.
An increasing reliance on traffic surveillance is needed for better vehicle detection at a wide-area. Automatic detection of vehicles in video surveillance data is a very challenging problem in computer vision with critical practical applications, such as traffic analysis and security. [1]
Meanwhile, artificial intelligence is giving surveillance cameras digital brains to match their eyes, letting them analyze live video with no humans necessary. It could be good news for public safety, helping police and first responders more easily spot crimes and accidents and have a range of scientific and industrial applications. [2]
Specifically, vehicle detection and counting are vital in computing traffic congestion on highways. Also, this information is needed by the city's command center for planning and decision-making tasks for the betterment of urban traffic flow and readiness to emergency events. The system has been developed to detect and count vehicles efficiently. Intelligent visual surveillance for road vehicles is a crucial component for developing autonomous smart transportation systems.
This paper presents an intelligent vehicle counting system, which is based from vehicle classification tasks. It comprised cumulative counting and targeted counting to provide insights to the city's traffic management office to solve traffic congestion in the highway and maximize the use of the command center's surveillance system.

Literature Review

Related studies have built embedded systems that can process the video frames immediately using Raspberry pi to count vehicles on the road. 75% MAP on an 80-20 train-test split and an accuracy rate between 70 to 90% were achieved in different experiments. Most frameworks used are the new YOLO and Tensorflow frameworks, which have built-in object-detection models. [1] [4] [5]
Another proposed method uses background subtraction technique to find foreground objects in a video sequence. To detect moving vehicles more accurately, several computer vision techniques, including thresholding, hole filling, and adaptive morphology operations, are then applied. For vehicle counting, using a virtual detection zone has been done. Experimental results show that the accuracy of the proposed vehicle counting system is around 96%. [3]
Real-time counting has also been proposed. One study reported the design and development of a novel, intelligent vehicle counting and classification system (iVCCS) that uses a wireless magnetometer sensor for real-time traffic surveillance. The other study proposed an automatic real-time background update algorithm for vehicle detection and an adaptive pattern for vehicle counting based on the virtual loop and detection line methods. [6] [7]

Methodology

A. Data Collection

Videos of CCTV footages were collected from the command center. A sample of four (4) videos that depicts entrance to the city has been taken as shown in Table I.

Table I Video Data for Training and Testing

Video Number	Entry Point	Description
1	Canlalay	The entry point from San Pedro
2	Soro-soro	The entry point from Carmona
3	Pavilion	The entry point from South Luzon Expressway
4	Platero	The entry point from Sta Rosa

Fig. 1 shows the map of the highway with the X marks as the position of CCTV cameras. In this busy highway, vehicles come from the neighbor cities. Using the newly-trained classifier, automatic classification and counting of vehicles is proposed.

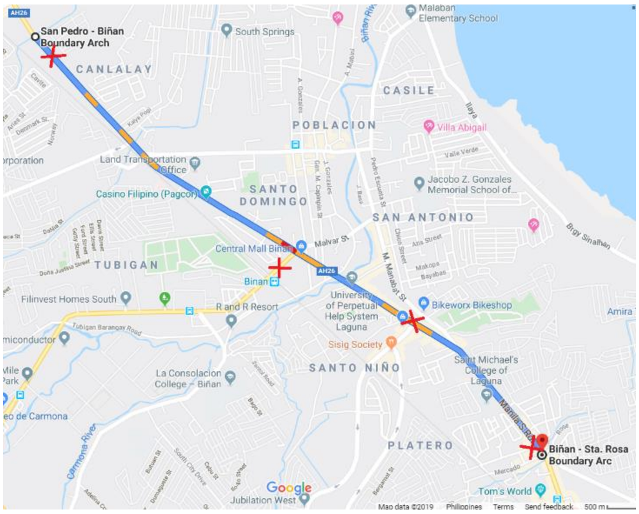


Fig. 1 The map of the highway under study

In addition, the CCTV footages portray different camera views as shown in Fig. 2. Notice that in frames 1, 2, and 3, the orientation of the vehicles is towards the left side of the frame, while video 4 is towards the right of the frame.

(a)

(b)

(c)

(d)

Fig. 2 Frames of Sample Videos: (a) frame of Video 1 (Canlalay); (b) frame of Video 2 (Sorosoro); (c) frame of Video 3 (Pavilion); (d) frame of Video 4 (Platero)

B. Image Pre-Processing

Videos were converted to images using Free Studio's Video to JPG converter that yielded to a total of 227 CCTV-captured images., these images were split to 80% (182 images) training dataset and 20% (45 images) testing dataset.

Table II Number of Labelled Objects for Training and Testing

Vehicle Class	Number of Labeled Objects for Training	Number of Labeled Objects for Testing
bicycle	14	5
bus	11	4
car	103	29

jeep	65	9
motorcycle	94	24
tricycle	88	33
truck	8	4
van	52	19
Total	**435**	**127**

Furthermore, these images were labeled based on the vehicle's eight (8) classes namely: bicycle, bus, car, jeep, motorcycle, tricycle, truck, and van. Also note that a single image file has a number of labelled objects.
Table II presents the number of labeled objects per vehicle class. Cars, motorcycles and tricycles are mostly the types of vehicles that pass along the roadways.

C. Training

After the dataset images has been properly labelled, the classifier has been built using Tensorflow 1.13 framework and the pre-trained model, Faster R-CNN Inception v2 COCO. Specifically, Tensorflow's object detection and counting APIs were the base design of the classifier and counter. [8] The training lasted for about four hours achieving a total loss of 0.04 which means that there is 0.04 difference between classifier's predicted output and the actual output.

D. Running the newly-trained model

A new set of videos from four (4) areas within the city were used to test the model for vehicle classification, detection and counting. Table III shows the details of the video for testing.

Table III Sample Videos for Training and Testing

Video Number	Entry Point	Duration (in seconds)	Frames per Second	Dimension
1	Canlalay	32	10	1280 x 790
2	Soro-soro	40	15.15	1280 x 960
3	Pavilion	32	10	1280 x 790
4	Platero	41	15	1382 x 750

E. Evaluation

The researchers used different metrics to measure the performance of the classification, detection and counting. Initially, they identified the four outcomes of the newly-trained classifier: true positive (TP), true negative (TN), false positive (FP) and false negative (FN); and built the confusion matrix as presented in Table IV.
Precision (equation 1), which is the percentage of correctly predicted classes out of all positive outcomes, and recall (equation 2), which is the percentage of correctly identified class out of the positive and negative outcomes, were also used. Additionally, F1 score (equation 3), which is the harmonic mean between precision and recall metrics, has been used to evaluate the classification performance of the newly-trained model while accuracy (equation 4) was used to measure the performance for vehicle counting.

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

$$F1\ Score = 2\ x\ \frac{Precision\ x\ Recall}{Precision + Recall} \quad (3)$$

$$Accuracy = \frac{Total\ number\ of\ counted\ vehicles}{Total\ number\ of\ vehicles}\ x\ 100 \quad (4)$$

Results and Discussion

A. Classification

Table IV presents the confusion matrix which shows the four outcomes of the newly-trained classifier. The actual values were compared to the machine predicted outcomes. In this matrix, the intersection of similar classes depicts the True Positive (TP) outcome. Given a car class, an example of a False Negative (green boxes) would be the intersection between van (actual) and car (predicted). The classifier predicted that there are two cars where there is actually no car. On the other hand, it is false positive (red boxes) if there are three actual cars but then the classifier says there's none.

Table IV Confusion Matrix after Running the Vehicle Classifier

		Machine Prediction							
		Bicycle	**Bus**	**Car**	**Jeep**	**Motorcycle**	**Tricycle**	**Truck**	**Van**
Actual Values	**bicycle**	**12**	0	0	0	0	2	0	0
	bus	0	**3**	0	0	0	0	0	1
	car	0	0	**33**	0	0	0	0	3
	jeep	0	0	0	**30**	0	0	2	1
	motorcycle	3	0	0	0	**45**	6	0	0
	tricycle	0	0	0	0	0	**28**	0	0
	truck	0	0	0	0	0	0	**2**	0
	van	0	0	2	4	0	0	0	**13**

Based on the confusion matrix, precision, recall and F1 score were computed to measure the performance of the classifier as presented in Table V. Tricycle and truck classes have the highest precision score which means that the classifier did not identify false positive outcomes in these classes. However, these classes also got 82% and 50% recall score which means that the classifier made incorrect predicted classes out of actual tricycles and trucks. Bus and motorcycle classes, on the other hand, have the highest recall scores but got 75% and 88% precision, respectively. Furthermore, the car class gained the highest F1 score which means that there is balance between its precision and recall scores. Generally, the newly-trained classifier has attained 85% F1-score which means it is a good model for vehicle classification.

Table V Performance Metric Scores for the Vehicle Classifier

Class	Precision	Recall	F1 Score
bicycle	86%	80%	83%
bus	75%	100%	86%
car	92%	94%	93%
jeep	91%	88%	90%
motorcycle	88%	100%	94%
tricycle	100%	82%	90%
truck	100%	50%	67%
van	68%	72%	70%
Average	**87%**	**83%**	**85%**

Vehicle Counting

Vehicles are identified based on the newly-trained classifier. In cumulative counting, detected vehicles are counted when they pass beyond region of interest (ROI) line (red line). ROI line is placed on entry points towards the national highway so that the command center can infer which entry points yield the highest number of vehicles. Fig. 3 indicates the cumulative count of vehicles which is 34. Likewise, it shows the detected vehicles inside bounding boxes with their corresponding class and scores. (e.g. tricycle 99%)

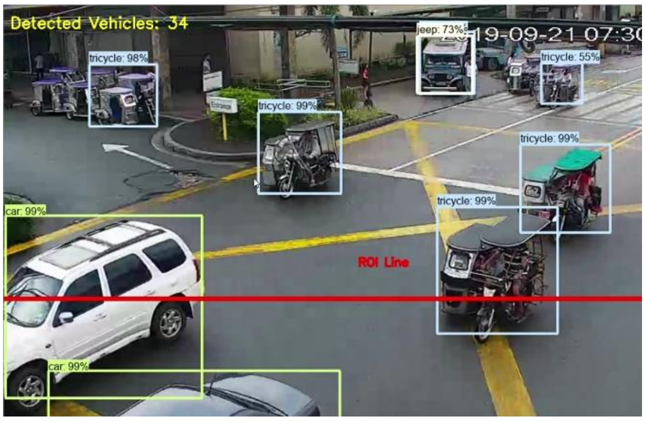


Fig. 3 Cumulative Counting of Vehicles

In targeted counting, all or selected vehicles are counted. Fig. 4(a) displays the count of all detected vehicles in the frame while Fig. 4(b) exhibits the count of car class only.

(a)

(b)

Fig. 4 Running the Counter program (a)cumulative counting; (b) targeted counting

Meanwhile, the classified vehicles are counted using the object counting API. Specifically, cumulative counting was applied. Since the ROI Line position is arbitrary, the researchers placed it in different positions (in pixel). It resulted to different counts as shown in Table VI. Notice that the yellow boxes represent nearly 100% counting accuracy provided that the ROI Line position is 550 in video 1, and 450 in videos 2 and 3. However, the counting result in video 4 obtained the lowest counting accuracy. Another factor might be the video's road and vehicle orientation towards the right of the frame as shown in Fig. 2d.

Table VI Count Results based on ROI Line Position

	Number of Vehicles Detected and Counted in Videos			
ROI Line Position (in pixel)	**1 46 vehicles**	**2 35 vehicles**	**3 38 vehicles**	**4 64 vehicles**
400	89	50	32	120
450	83	32	37	95
500	59	43	31	87
550	45	43	34	70
600	18	25	40	53

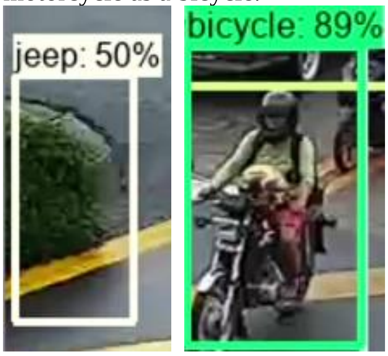
The final counting results with selected ROI Line position is presented in Table VII.

Table VII Final Count Results

Video No.	**Selected ROI Line Position**	**Number of Vehicles**	**Number of Counted Vehicles**	**Accuracy (%)**
1	550	46	45	98%
2	450	35	32	91%
3	450	38	37	97%
4	550	64	57	89%
Total	**N/A**	**183**	**171**	**94%**

Based on the performance of the classifier and counter, they have shown good results with 85% F1 score and 94% accuracy, respectively. However, there are still errors that been encountered. Errors in classification includes, misclassified objects (non-vehicle) and (vehicle but different class) as shown in Fig. 5. The classifier predicted with 50%

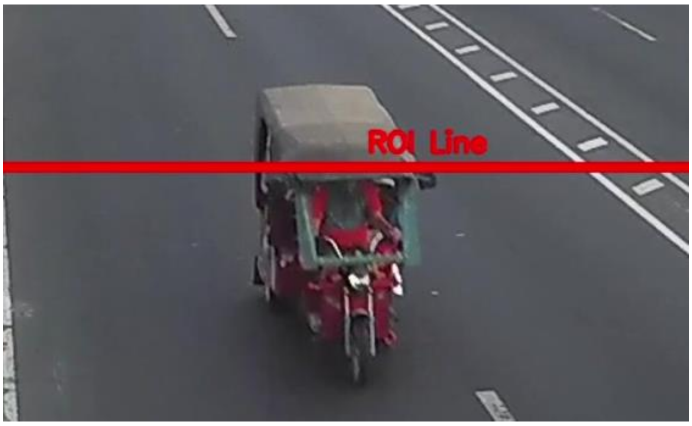
score that the bush is a jeep and it misclassified the motorcycle as a bicycle.



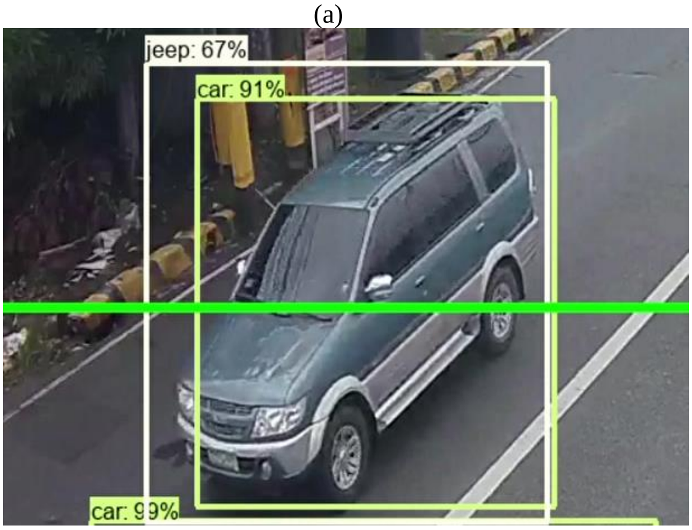
(a) (b)

Fig. 5 Vehicle Classification Errors (a) detection of non-vehicle object; (b) misclassification of vehicle class

Errors in counting include vehicles not counted because the object is not detected and vehicles are counted more than once because of object split as shown in Fig. 6. Notice that in Fig. 6a, the tricycle does not have a bounding box which means that it is not detected by the classifier, therefore it is not counted as indicated by the red ROI line. On the other hand, Fig. 6b illustrates that the car which has two bounding boxes is counted twice as indicated by the green ROI line.



(a)



b)

Fig. 6 Vehicle Counting Errors (a) not counted vehicle; (b) vehicle counted more than once

Cumulative counting can be applied in traffic analysis in video surveillance systems. Targeted counting can be used in counting specific vehicle class for revisiting road policies and planning for a better and intelligent transport system.

Conclusion and Recommendations

Conclusion: Based on the results of the performance of vehicle classifier and counter, the following conclusions were drawn:

- Intelligent counting is based on the performance of detection and classification model. If the classifier detected the vehicle correctly, counting will perform well also. This is evident in the F1-score of 85% in classifying vehicle objects and 94% accuracy in vehicle counting. However, detection errors are still present. One of the leading error is that the classifier had counted the same vehicle twice or counted a non-vehicle object that resulted in over-counting.
- Fine tuning the ROI line position is significant in the performance of counter. If we set the ROI line in the upper part of the frame, it may count far-away misclassified vehicles or non-vehicle objects.

Recommendations: Based on the conclusions stated, the following recommendations are suggested:

- The classifier can be further improved by adding more labelled images and increasing the threshold greater than the default 50% in detecting a specific vehicle class. Through this, detection errors can be minimized towards an improved intelligent counting.
- In labelling images for training, the distance of vehicle should be based on ROI line. Putting labels on far-away vehicles is not necessary for cumulative counting since the classifier should detect the features of vehicles closely.
- The design of two regions of interest line may be proposed to distinguish between entry and exit vehicles.
- The integration of the vehicle classification, detection and counting is recommended for intelligent surveillance systems.

References

[1] M. V. N. H. Kumar and B. Vasanth, “Vehicle Detection , Tracking and Counting Objects For Traffic Surveillance System Using Raspberry-Pi,” Int. J. Mod. Trends Eng. Res., pp. 233–242, 2015.

[2] J. Vincent, “Artificial Intelligence is Going to Supercharge Surveillance,” The Verge, pp. 1–7, 2018.

[3] N. Seenouvong, U. Watchareeruetai, C. Nuthong, K. Khongsomboon, and N. Ohnishi, “A computer vision based vehicle detection and counting system,” in 2016 8th International Conference on Knowledge and Smart Technology, KST 2016, 2016.

[4] M. S. Chauhan, A. Singh, M. Khemka, A. Prateek, and R. Sen, "Embedded CNN based vehicle classification and counting in non-laned road traffic," ACM Int. Conf. Proceeding Ser., vol. 1, 2019.

[5] M. Bhavsar, M. Parhadkar, P. Gyanchandani, A. Tiwari, and A. Nilawar, "Portable Device for Vehicle Counting , Classification and Speed," Int. J. Adv. Res. Innov. Ideas Educ., vol. 5, no. 2, pp. 813–819, 2019.

[6] W. Balid, H. Tafish, and H. H. Refai, "Intelligent Vehicle Counting and Classification Sensor for Real-Time Traffic Surveillance," IEEE Trans. Intell. Transp. Syst., vol. 19, no. 6, pp. 1784–1794, 2018.

[7] F. Liu, Z. Zeng, and R. Jiang, "A video-based real-time adaptive vehicle-counting system for urban roads," PLoS One, vol. 12, no. 11, pp. 1–16, 2017.

[8] A. Ozlu, "Vehicle Detection, Tracking and Counting by TensorFlow," Github Repository, 2018. [Online]. Available: https://github.com/ahmetozlu/vehicle_counting_tensorflow.

Authors Profile

Maria Crystal Orozco She is a college professor and has been teaching college students since 2007 in her alma mater, University of Perpetual Help System wherein she completed her Bachelor of Science in Computer Science (BSCS) degree. Consequently, she finished her Master's degree in Information Technology (MIT) at Manuel S. Enverga University Foundation. Currently, she is a graduate student of Technological Institute of the Philippines - Manila pursuing Doctor in Information Technology. Through this, she had opportunities in managing IT projects and attending research conferences. The most notable one was when she presented her paper in South Korea. From then on, she has been interested in data mining and data science, image processing and computer vision, and the recently-discovered deep learning, a subset of artificial intelligence.

Corazon Rebong She is currently the dean of School of Computer Studies and Technology of Colegio de San Juan de Letran and a part-time graduate school professor of the Technological Institute of the Philippines – Manila. She completed her BS Computer Engineering in Adamson University and accomplished two Master's degree programs, namely: Master in Business and Administration (MBA) which she completed in Letran Calamba; and Master of Engineering major in Computer Engineering (MEngCoE) in Pamantasan ng Lungsod ng Maynila. Additionally, she attained her Doctor of Philosophy in Management major in Information Technology Management (PhD- ITM), and received the Merretissimus Award for her dissertation in 2009. From then on, she continues to contribute to quality education by being a PACUCOA accreditor, ISO Auditor and RQAT Member of different quality organizations. Moreover, she has a number of published papers authored, not to mention, a book co-authored about Ethical and Legal Issues in a Wired World". Furthermore, she is affiliated to national organizations of the ITE and Engineering fields, such as ICpEP, PSITE 4 and PhiCES, as officer and/or member.