Towards Real-time Traffic Flow Estimation using YOLO and SORT from Surveillance Video Footage

Nilani Algiriyage*
Massey University
r.nilani@massey.ac.nz

Raj Prasanna
Massey University
r.prasanna@massey.ac.nz

Emma E H Doyle
Massey University
e.e.hudson-doyle@massey.ac.nz

Kristin Stock
Massey University
k.stock@massey.ac.nz

David Johnston
Massey University
d.m.johnston@massey.ac.nz

Minura Punchihewa
University of Kelaniya
minurapunchihewa17@gmail.com

Santhoopa Jayawardhana
University of Kelaniya
santhoopa@gmail.com

ABSTRACT

Traffic emergencies and resulting delays cause a significant impact on the economy and society. Traffic flow estimation is one of the early steps in urban planning and managing traffic infrastructure. Traditionally, traffic flow rates were commonly measured using underground inductive loops, pneumatic road tubes, and temporary manual counts. However, these approaches can not be used in large areas due to high costs, road surface degradation and implementation difficulties. Recent advancement of computer vision techniques in combination with freely available closed-circuit television (CCTV) datasets has provided opportunities for vehicle detection and classification. This study addresses the problem of estimating traffic flow using low-quality video data from a surveillance camera. Therefore, we have trained the novel YOLOv4 algorithm for five object classes (car, truck, van, bike, and bus). Also, we introduce an algorithm to count the vehicles using the SORT tracker based on movement direction such as "northbound" and "southbound" to obtain the traffic flow rates. The experimental results, for a CCTV footage in Christchurch, New Zealand shows the effectiveness of the proposed approach. In future research, we expect to train on large and more diverse datasets that cover various weather and lighting conditions.

Keywords

Computer vision, traffic flow, YOLOv4, CCTV big data.

INTRODUCTION

Today, with the high rate of urbanization, the number of vehicles in an urban road network has increased significantly. According to statistics released by the Ministry of Transport, there were 11,449 accidents in New Zealand, including 2,449 that caused serious and fatal injuries in 2019[1]. The resulting congestion and related issues after crash incidents cause substantial economic loss and disrupt the community's everyday life. Furthermore, other natural

*corresponding author
[1] https://www.transport.govt.nz/statistics-and-insights/safety-annual-statistics/

and man-made disasters such as flooding, landslides and terrorist attacks causing traffic emergencies are inevitable. During such emergencies, the road network becomes congested, making evacuation impossible and rescue personnel and supplies unable to be transported (Alam et al. 2018; Mostafizi et al. 2019). Therefore, traffic emergencies must be addressed in an intelligent transport system to ensure secure, responsive and efficient transportation for everyone (Peppa et al. 2018; Fedorov et al. 2019).

Understanding road traffic behaviour is a key component of an emergency traffic response plan. Traffic flow estimation is the first step for identifying the road traffic patterns, contributing to traffic modelling, urban planning and design processes for all aspects of a road network (Fedorov et al. 2019). Traffic data acquisition is typically performed using underground inductive-loops, pneumatic road tubes, and manual counts. However, these methods are labour intensive, expensive, difficult to install and can be inaccurate. Also, they could damage the road surface and reduce the quality and life of the road and thus can not be used in large areas (Algiriyage et al. 2020).

Closed-circuit television (CCTV) systems are now increasingly popular and are installed in many public places to enable real-time surveillance. As these systems are continuously operated, they generate a vast amount of data that contribute to big data. Recent developments in computer vision research have heightened the need for using CCTV images to tackle practical problems such as traffic congestion detection (Kurniawan et al. 2018), automatic licence plate recognition (Indira et al. 2019; Laroca et al. 2018), emergency vehicle detection (Roy and Rahman 2019) and accident detection (Ijjina et al. 2019; Veni et al. 2020). However, traffic flow estimation using computer vision algorithms for surveillance camera datasets is still in very early development. Difficulties in moving, storing, processing and developing efficient algorithms to analyse CCTV data have been identified as significant challenges (Fedorov et al. 2019).

This study aims to answer the research question: 1) Can traffic flow be estimated from low-quality CCTV video footage in real-time?. As a case study, we focus on a multi-lane road in Christchurch Central Business District (CBD). We obtain the traffic flow based on vehicle movement direction such as "northbound" and "southbound". Furthermore, vehicle counts are obtained for five vehicle classes, such as *car*, *bus*, *van* and *truck* and *bike*. We train You Only Look Once (YOLOv4) algorithm (Bochkovskiy et al. 2020) for vehicle detection and classification and Simple Online and Real-time Tracking (SORT) (Bewley et al. 2016) algorithm for vehicle tracking. Last year, we introduced our algorithm as a conference poster[2]. However, it was an early in-progress work that we used YOLOv3, trained on Common Objects in Context (COCO) dataset for four vehicle classes. The algorithm discussed in this paper is improved by custom training YOLOv4. Authorities can use our algorithm for traffic flow monitoring, traffic anomaly identification, and the development of emergency rescue plans. Also, responders are able to make management decisions such as detour allocation and changing traffic light timing length during emergencies by using real-time traffic flow.

The contributions of the paper are as follows:

- We have trained YOLOv4 with our own vehicle image dataset and publicly made available the dataset for future researchers [3].
- We have introduced an algorithm to count directional traffic flow using YOLOv4 and SORT tracker.
- We show that the custom trained YOLOv4 performs well having a F1-score of more than 0.95 for car class during day, evening and night times using a low-frame-rate footage.

The rest of our paper is outlined as follows. The section Related Work reviews the existing work. Then in the Methodology section, we illustrate the architecture and algorithms implemented. The Results section describes our research findings. Finally, we present concluding remarks and future research steps in section the Discussion.

RELATED WORK

Recently, visual datasets obtained from surveillance cameras and aerial vehicles have been explored for many traffic monitoring applications (Im et al. 2016; Zhang et al. 2017; Ke et al. 2018; Agarwal et al. 2020). Convolutional Neural Networks (CNN) based object detectors have been widely adopted for such visual datasets in computer vision research. These algorithms can generally be divided into two major groups, namely, single-stage detectors and two-stage detectors. Single-stage detectors such as Single Shot Detector (SSD) and YOLO are generally fast and predict object bounding boxes together with classes within a single network pass (Redmon, Divvala, et al.

[2] https://conference.eresearch.edu.au/2020/09/real-time-traffic-flow-estimation-based-on-deep-learning-using-cctv-videos/

[3] Annotated vehicle dataset, Traffic_Flow_Estimation

2016; Liu et al. 2016). In contrast, two-staged detection happens in two stages. First, the model proposes a set of regions of interests by selective search or using Regional Proposal Network (RPN). Then a classifier only processes the region candidates to identify the objects (Uijlings et al. 2013; Girshick et al. 2014; Girshick 2015; Ren et al. 2015; He et al. 2017) (see Fig. 1). As a result, two-stage detection tends to be slow (e.g., R-CNN family networks including the original R-CNN, Fast R- CNN, Faster R-CNN and Mask R-CNN).

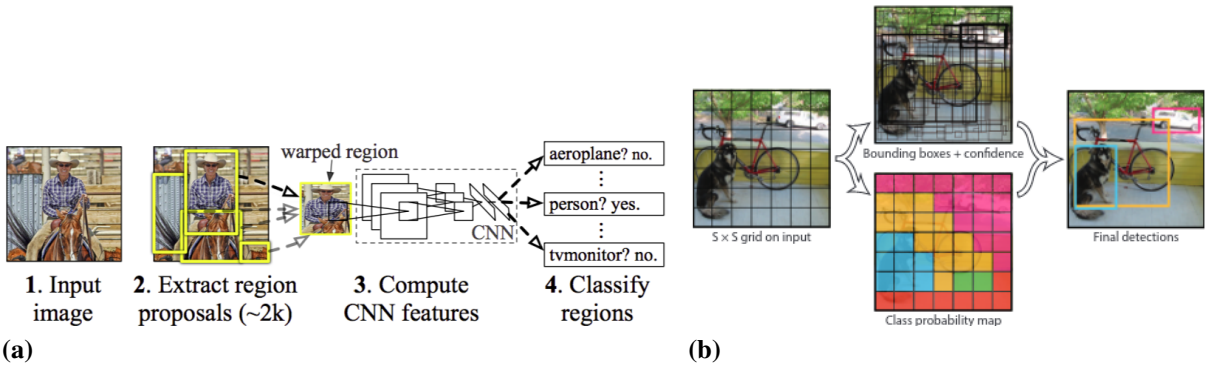


Figure 1. Two-stage detection vs single-stage detection (a) R-CNN architecture (Girshick et al. 2014) (b) YOLO object detection (Redmon, Divvala, et al. 2016).

Vehicle object detection, classification and tracking are the three main tasks involved while processing video datasets for traffic flow estimation (Fedorov et al. 2019; Oltean et al. 2019). Object detection deals with drawing bounding boxes around the objects of interest to locate it within the image. Classification helps to categorise objects into different classes such as "car, bus, truck". In 2015, Redmon et al. introduced You Only Look Once (YOLO) as a fast, accurate and real-time object detection system. It went through several modifications of the architecture until it produced YOLOv3 in 2018 (Redmon and Farhadi 2017; Redmon and Farhadi 2018). Chakraborty at el. (2018) evaluated the performance of deep convolution neural network (DCNN), support vector machine (SVM) and basic YOLO algorithm for classifying traffic congestion from CCTV images. They show that YOLO algorithm obtaining the highest accuracy of 91.4 for the task. In a similar study by Algiriyage et al. (2020) uses a CCTV image dataset to obtain traffic flow and compare the performance of YOLOv3, faster R-CNN and mask-RCNN for object detection. They show that among them, YOLOv3 showed the best performance in terms of speed and accuracy for their image dataset having a precision value of 0.96. Corovic at el. (Corovic et al. 2018) train YOLOv3 to detect five classes of objects, namely, cars, trucks, pedestrians, traffic signs and traffic lights under different lighting conditions. Though they used a small dataset having 300 images, they could get an F1-score of 0.59. YOLOv3-tiny version pre-trained on COCO dataset was used by Oltean at el. (2019) for real-time traffic counting. They show that for the few frames they considered for the experiment out of the total 27, 26 vehicles were correctly detected. In 2020, YOLOv4 was introduced as a faster and more accurate detector than the all available CNN based detectors (Wang and Liao 2020). However, far too little attention has been paid to research on using YOLOv4 for vehicle object detection.

Object tracking analyses the movement path of an object across different frames. Depending on the tracking target, there are two categories of tracking algorithms such as Single object tracking (SOT) and Multiple object tracking (MOT). In SOT, a single object is tracked from the beginning, while in MOT, several objects are detected and tracked from one frame to the other (Nam Bui et al. 2020). Two well-known examples of SOT algorithms include Kalman Filtering and Particle Filtering, whereas SORT and DeepSORT are two state-of-the-art MOT algorithms (Bui et al. 2020). Several studies have investigated vehicle tracking and counting from CCTV videos (Choudhury et al. 2017; Lucking et al. 2020; Chakraborty et al. 2018; Shaweddy and Wahyono 2019). For example studies by, Bui et al. (2020) and Nam Bui et al. (2020) use DeepSORT for vehicle tracking and virtual lines for traffic counting. However, the direction of vehicle movement is not considered while obtaining the traffic flow. Closer to our objective is the traffic counting system introduced by Fedorov et al. (Fedorov et al. 2019). They use Faster-RCNN object detector and SORT tracker. However, they have carried out experiments for 982 video frames and do not obtain the traffic flow by vehicle class. Apart from Fedorov (2019), there is a general lack of research in investigating the real-time traffic flow estimation from surveillance video, while also considering movement direction and vehicle class. Thus, this study was set out to explore traffic flow estimation in real-time from CCTV video considering these gaps.

METHODOLOGY

This study investigates real-time traffic flow estimation from low-quality surveillance video data. Also, we classify vehicles and obtain the flow rate based on their movement directions. In order to achieve this objective, we train the

novel YOLOv4 algorithm with a custom image dataset collected from the same camera to detect five object classes namely, car, bus, truck, van and bike. Then the trained weights are used for the traffic flow estimation module (see Figure 2). The traffic flow estimation module counts vehicles based on the direction of movement and the class of the vehicle from CCTV video data. Therefore, this module is divided into three sub-tasks: vehicle detection, vehicle tracking, and traffic flow estimation. The vehicle detection module draws a bounding box around vehicle objects in order to locate it within a frame, while the vehicle tracking module tracks the movement of a vehicle object between different frames. Our algorithm can be easily applied to any similar location with very few modifications and extended to complex locations with changes based on the degree of complexity.

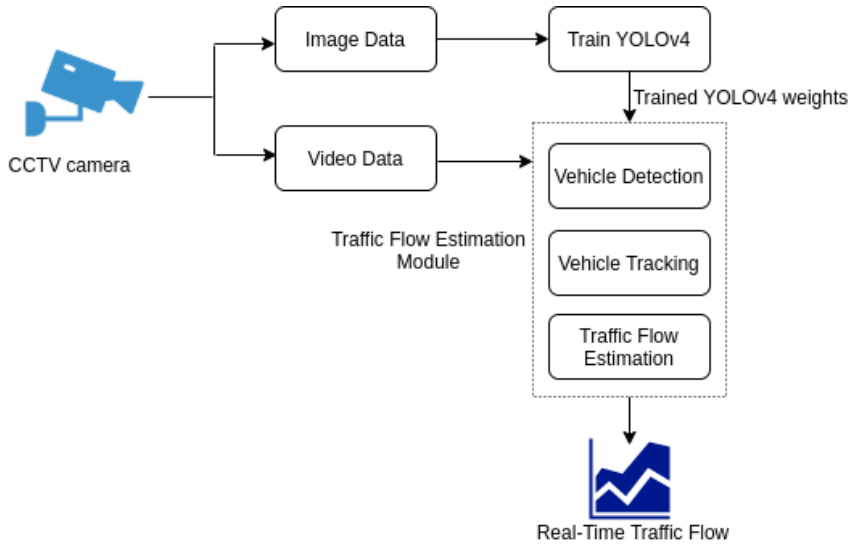


Figure 2. Methodology for Real-time traffic flow estimation.

Dataset

We obtained CCTV image and video datasets from the New Zealand Transport Agency (NZTA), Christchurch, New Zealand. As a case study, we selected a busy road namely "West along Yaldhurst Rd from Curletts Rd" in Christchurch CBD. The image datasets were used to train YOLOv4 while the footage datasets were used to validate the real-time traffic flow counting algorithm. The camera at the selected location generates video with a frequency of ≈ 10 frames per second (fps) and resolution of $1280 * 720$ $(width * height)$. The three video footage that we analysed for this research was recorded during the day, evening and night times, in February 2020. Table 1 and Table 2 summarise the details of the image and video dataset respectively.

Table 1. Details of the image dataset used to train YOLOv4

Vehicle Class	Total Instances
Car	13,627
Bus	141
Van	779
Truck	1,273
Bike	280

Table 2. Details of the analysed CCTV videos (hr: hours, mins: minutes and secs: seconds)

Description	Start Time	Finish Time	Duration	No of frames
Video 01	10:00:00 (UTC + 12:00)	11:42:33 (UTC + 12:00)	1 hr, 42 mins & 33 secs	69, 676
Video 02	18:06:17 (UTC + 12:00)	19:06:56 (UTC + 12:00)	1 hr, 0 mins & 39 secs	86, 987
Video 03	20:26:12 (UTC + 12:00)	21:56:30 (UTC + 12:00)	1 hr, 30 mins & 18 secs	55, 794

Vehicle detection

The foundation of our detection module is the novel single-stage YOLOv4 detector (Bochkovskiy et al. 2020). This model was trained on the image dataset obtained from the NZTA as described in Table1, using the Darknet 1[4] implementation of the YOLOv4 algorithm. The images were annotated using LabelImg tool[5] prior to carrying out the training process. The training was carried out using the Mahuika High-Performance Computing (HPC) cluster of the New Zealand eScience Infrastructure (NeSI) for a total of 10,000 epochs. The total amount of time taken for the training was around 15 hours on 2 GPU cores.

Vehicle tracking

Vehicle tracking deals with identifying the vehicle movement from one frame to the other. To handle this, we adopted SORT tracker (Bewley et al. 2016) as it is both powerful and fast (Fedorov et al. 2019).

Vehicle movement direction estimation and traffic flow counting

Figure. 3 shows a drawing of the location we analysed which is a "multi-lane" road where there are two lanes for each direction. The *traffic flow rate* can be defined as the number of vehicles during the tth time interval at the ith observation location in a transportation network which is given by Eq. 1 (Kumarage et al. 2017).

$$X_i^t = n/t \tag{1}$$

where:
X_i^t = traffic flow rate
n = number of vehicles
t = time duration

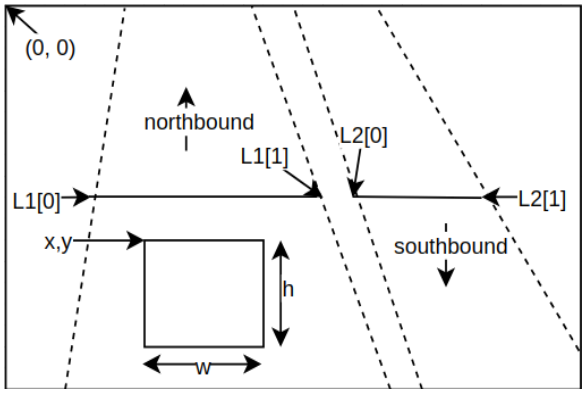


Figure 3. Drawing of the location analysed - Line coordinates (L1[0], L1[1], L2[0], L2[1]) and bounding box properties of a vehicle object (x, y, width (w), height (h)).

The width and height of a single frame in the analysed video is $1280 * 780$. We define two lines with coordinates $(400, 300) - L1[0]$, $(750, 300) - L1[1]$, $(820, 300) - L2[0]$ and $(1160, 300) - L2[1]$ to identify the movement direction of a vehicle such as "northbound" and "southbound". We analyse each frame (i) in the set of frames (I) of the footage. If a vehicle enters a particular line, it is detected, classified and tracked over different frames. A simple mathematical calculation is applied to count the intersections between the vehicles' previous and current frame positions using the defined lines. This is performed using the center of the bounding boxes ($cnt0$, $cnt1$) in the current and previous positions and also using the line coordinates. Then, when an intersection is found, our algorithm checks the YOLOv4 class label to increase the car (n_car_count), bus (n_bus_count), van (n_van_count), truck (n_truck_count) and bike (n_bike_count) count in each movement direction. Our algorithm writes the real-time traffic counts into a text file. We use python pandas library [6] and and matplotlib FuncAnimation [7] to live plot the traffic flow. The pseudo-code for the traffic estimation is proposed in Algorithms [8] and Algorithm [9].

[4]Darknet, open source neural network framework, https://github.com/pjreddie/darknet
[5]LabelImg, graphical image annotation tool, https://github.com/tzutalin/labelImg
[6]pandas software library, https://pandas.pydata.org/
[7]Real-time plotting library, https://matplotlib.org/stable/api/_as_gen/matplotlib.animation.FuncAnimation.html
[8]Traffic flow estimation algorithm, Traffic-Flow-from-Footage/blob/master/Traffic_Flow_Estimation.png
[9]Intersection detection algorithm, Traffic-Flow-from-Footage/blob/master/Intersect.png

RESULTS

The overall Mean Average Precision (mAP) of the YOLOv4 model was 92.35%, and the performance on each class as per the validation dataset is summarized in Table 3.

Table 3. Man Average Precision (mAP) of vehicle detector classes

Class	Average Precision (AP)
Car	96.94%
Bus	93.64%
Van	90.22%
Truck	90.24%
Bike	90.72%

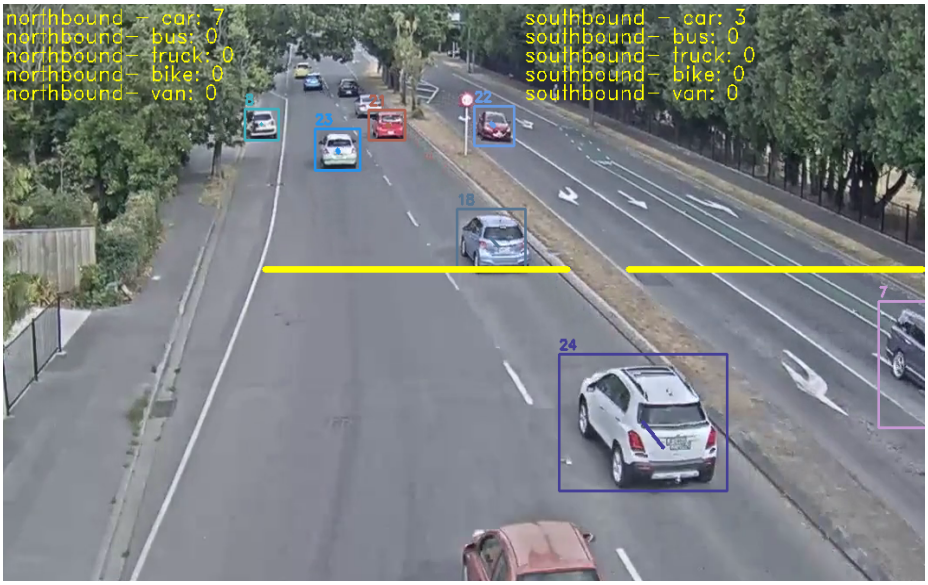


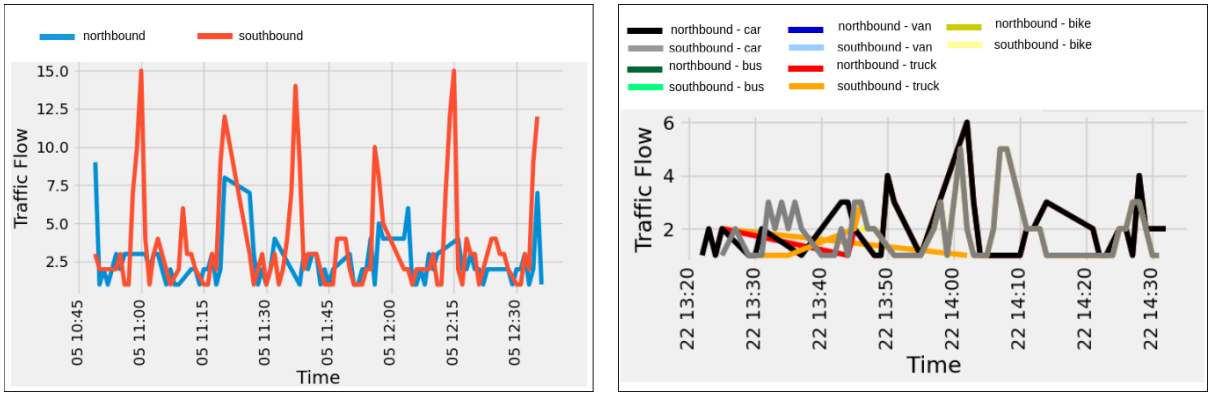
Figure 4. Traffic flow estimation from video footage.

The AP value has dropped proportionately to the number of instances we used in the training dataset. For example, we used a higher number of images for the car class and hence it has got an AP value of 96.94%. Figure 4 illustrates the real-time traffic flow estimation system based on custom trained YOLOv4. Two live plots are generated to show the directional traffic flow, and the flow counts by vehicle class, as presented in Figure 5.

To measure the detection accuracy, we manually counted the number of vehicles in each class for the video footage analysed and used as ground-truth values. Then the accuracy value is measured using the equation 2.

$$\text{Accuracy} = \frac{\text{No of correct detections}}{\text{No of ground-truth detections}} \tag{2}$$

Table 4 illustrates the accuracy scores for all three videos considered for our experiments. The mean accuracy for obtaining "northbound" traffic flow is 0.7114 while the "southbound" is 0.6397 for all vehicle classes. A possible explanation for this might be that the camera was located close to the "northbound" lane. Therefore, our detection module could identify vehicles in the "northbound" lane more accurately. Furthermore, the flow count for the car object class is more accurate; obtaining a mean accuracy score of 0.9595. For the location we considered, the vast majority of the vehicles consisted of car class. The higher accuracy for car class indicates that our system performs well finding the traffic flow from video footage. However, the detection accuracy for bus class is lower (mean accuracy score : 0.1667). An implication of this is the possibility that the lower number of bus image instances in the training dataset. Furthermore, it is interesting to note that there is no significant difference between the accuracy scores of the three footage considered during different times of the day. Finally, several limitations need to be considered. First, we didn't incorporate the vehicle re-identification problem. For instance, the same vehicle



(a) **(b)**

Figure 5. Live plots of traffic flow (a) Directional traffic flow (b) Traffic flow by vehicle class.

Table 4. Number of vehicles counted by humans (ground-truth), automatically by our algorithm and the accuracy for video 01 (day), video 02 (evening) and video 03 (night).

Video Dataset	Vehicle Class	Ground-truth		Number Detections		Accuracy	
		northbound	southbound	northbound	southbound	northbound	southbound
Video 01	car	1832	1308	1770	1189	0.9662	0.9090
	bus	8	4	5	2	0.6250	0.5000
	van	40	58	28	34	0.7000	0.5862
	truck	343	389	220	248	0.6414	0.6375
	bike	2	2	1	0	0.5000	0.0000
Video 02	car	1396	1057	1368	1022	0.9799	0.9669
	bus	3	2	2	1	0.6667	0.5000
	van	20	18	14	12	0.7000	0.6667
	truck	34	26	24	18	0.7059	0.6923
	bike	2	0	1	0	0.5000	1.0000
Video 03	car	798	802	774	774	0.9699	0.9651
	bus	3	2	1	0	0.3333	0.0000
	van	22	17	18	12	0.8182	0.7059
	truck	68	46	58	38	0.8529	0.8261
	bike	2	1	1	0	0.5000	0.0000

can be counted many times with the current approach. This can affect the traffic flow count as duplicated entries. Second, our CCTV footage was captured during the summertime in New Zealand. The lighting conditions might vary during other times of the year. Future work needs to explore the detection accuracy during night times. Third, we lose tracking vehicle objects in some frames due to occasional poor quality visuals generated from the cameras. As a result, some vehicle objects are missed by the flow counting algorithm.

CONCLUSION

In this study, we focused on obtaining real-time traffic flow using low-quality CCTV footage. As a case study, we selected one of the busiest multi-lane roads in Christchurch CBD, New Zealand. We trained the YOLOv4 model to detect five vehicle object classes: car, bus, van, truck and bike. The test results of this study show that we could obtain a high accuracy for the car class (mean accuracy score : 0.9595) while obtaining the traffic flow based on the movement direction. However, as our training image dataset was unbalanced, we obtained a lower accuracy score for the bus class. Therefore, further work needs to train YOLOv4 with a higher number of vehicles for low accurate classes. Therefore, in future work, we hope to train YOLOv4 algorithm with a large and diverse dataset. In addition, we hope to apply this work to more complex crossroads and consider the traffic count per lane.

REFERENCES

Agarwal, A., Rana, H., Vats, V., and Saraswat, M. (2020). "Efficient Traffic Density Estimation Using Convolutional Neural Network". In: *2020 6th International Conference on Signal Processing and Communication (ICSC)*. IEEE, pp. 96–100.

Alam, M. J., Habib, M. A., Quigley, K., and Webster, T. L. (2018). "Evaluation of the traffic impacts of mass evacuation of Halifax: flood risk and dynamic traffic microsimulation modeling". In: *Transportation research record* 2672.1, pp. 148–160.

Algiriyage, N., Prasanna, R., Doyle, E. E., Stock, K., and Johnston, D. (2020). "Traffic Flow Estimation based on Deep Learning for Emergency Traffic Management using CCTV Images". In: *17th International Conference on Information Systems for Crisis Response and Management*, pp. 100–109.

Bewley, A., Ge, Z., Ott, L., Ramos, F., and Upcroft, B. (2016). "Simple online and realtime tracking". In: *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 3464–3468.

Bochkovskiy, A., Wang, C.-Y., and Liao, H.-Y. M. (2020). "YOLOv4: Optimal Speed and Accuracy of Object Detection". In: *arXiv preprint arXiv:2004.10934*.

Bui, K.-H. N., Yi, H., Jung, H., and Cho, J. (2020). "Video-Based Traffic Flow Analysis for Turning Volume Estimation at Signalized Intersections". In: *Asian Conference on Intelligent Information and Database Systems*. Springer, pp. 152–162.

Chakraborty, P., Adu-Gyamfi, Y. O., Poddar, S., Ahsani, V., Sharma, A., and Sarkar, S. (2018). "Traffic Congestion Detection from Camera Images using Deep Convolution Neural Networks". In: *Transportation Research Record* 2672.45, pp. 222–231.

Choudhury, S., Chattopadhyay, S. P., and Hazra, T. K. (2017). "Vehicle detection and counting using haar feature-based classifier". In: *2017 8th Industrial Automation and Electromechanical Engineering Conference, IEMECON 2017*, pp. 106–109.

Corovic, A., Ilic, V., Duric, S., Marijan, M., and Pavkovic, B. (2018). "The Real-Time Detection of Traffic Participants Using YOLO Algorithm". In: *2018 26th Telecommunications Forum, TELFOR 2018 - Proceedings*.

Fedorov, A., Nikolskaia, K., Ivanov, S., Shepelev, V., and Minbaleev, A. (2019). "Traffic flow estimation with data from a video surveillance camera". In: *Journal of Big Data* 6.1, p. 73.

Girshick, R. (2015). "Fast r-cnn". In: *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). "Rich feature hierarchies for accurate object detection and semantic segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580–587.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). "Mask r-cnn". In: *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969.

Ijjina, E. P., Chand, D., Gupta, S., and Goutham, K. (2019). "Computer Vision-based Accident Detection in Traffic Surveillance". In: *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. IEEE, pp. 1–6.

Im, H., Hong, B., Jeon, S., and Hong, J. (2016). "Bigdata analytics on CCTV images for collecting traffic information". In: *2016 International Conference on Big Data and Smart Computing (BigComp)*. IEEE, pp. 525–528.

Indira, K., Mohan, K., and Nikhilashwary, T. (2019). "Automatic license plate recognition". In: *Recent Trends in Signal and Image Processing*. Springer, pp. 67–77.

Ke, R., Li, Z., Tang, J., Pan, Z., and Wang, Y. (2018). "Real-time traffic flow parameter estimation from UAV video based on ensemble classifier and optical flow". In: *IEEE Transactions on Intelligent Transportation Systems* 20.1, pp. 54–64.

Kumarage, S. P., Rajapaksha, R., De Silva, D., and Bandara, J. (2017). "Traffic Flow Estimation for Urban Roads Based on Crowdsourced Data and Machine Learning Principles". In: *First International Conference on Intelligent Transport Systems*. Springer, pp. 263–273.

Kurniawan, J., Syahra, S. G., Dewa, C. K., et al. (2018). "Traffic Congestion Detection: Learning from CCTV Monitoring Images using Convolutional Neural Network". In: *Procedia computer science* 144, pp. 291–297.

Laroca, R., Severo, E., Zanlorensi, L. A., Oliveira, L. S., Gonçalves, G. R., Schwartz, W. R., and Menotti, D. (2018). "A robust real-time automatic license plate recognition based on the YOLO detector". In: *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, pp. 1–10.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016). "Ssd: Single shot multibox detector". In: *European conference on computer vision*. Springer, pp. 21–37.

Lucking, M., Rivera, E., Kohout, L., Zimmermann, C., Polad, D., and Stork, W. (2020). "A video-based vehicle counting system using an embedded device in realistic traffic conditions". In: *IEEE World Forum on Internet of Things, WF-IoT 2020 - Symposium Proceedings*, pp. 1–6.

Mostafizi, A., Wang, H., and Dong, S. (2019). "Understanding the multimodal evacuation behavior for a near-field tsunami". In: *Transportation research record* 2673.11, pp. 480–492.

Nam Bui, K. H., Yi, H., and Cho, J. (2020). "A multi-class multi-movement vehicle counting framework for traffic analysis in complex areas using CCTV systems". In: *Energies* 13.8.

Oltean, G., Florea, C., Orghidan, R., and Oltean, V. (2019). "Towards Real Time Vehicle Counting using YOLO-Tiny and Fast Motion Estimation". In: *SIITME 2019 - 2019 IEEE 25th International Symposium for Design and Technology in Electronic Packaging, Proceedings* October, pp. 240–243.

Peppa, M., Bell, D., Komar, T., and Xiao, W. (2018). "URBAN TRAFFIC FLOW ANALYSIS BASED ON DEEP LEARNING CAR DETECTION FROM CCTV IMAGE SERIES." In: *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences* 42.4.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). "You only look once: Unified, real-time object detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788.

Redmon, J. and Farhadi, A. (2017). "YOLO9000: better, faster, stronger". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7263–7271.

Redmon, J. and Farhadi, A. (2018). "Yolov3: An incremental improvement". In: *arXiv preprint arXiv:1804.02767*.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). "Faster r-cnn: Towards real-time object detection with region proposal networks". In: *Advances in neural information processing systems*, pp. 91–99.

Roy, S. and Rahman, M. S. (2019). "Emergency Vehicle Detection on Heavy Traffic Road from CCTV Footage Using Deep Convolutional Neural Network". In: *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*. IEEE, pp. 1–6.

Shaweddy, V. and Wahyono, W. (2019). "Vehicle counting framework for intelligent traffic monitoring system". In: *Proceedings - 2019 5th International Conference on Science and Technology, ICST 2019*, pp. 1–5.

Uijlings, J. R., Van De Sande, K. E., Gevers, T., and Smeulders, A. W. (2013). "Selective search for object recognition". In: *International journal of computer vision* 104.2, pp. 154–171.

Veni, S., Anand, R., and Santosh, B. (2020). "Road Accident Detection and Severity Determination from CCTV Surveillance". In: *Advances in Distributed Computing and Machine Learning*. Springer, pp. 247–256.

Wang, C.-y. and Liao, H.-y. M. (2020). "YOLOv4: Optimal Speed and Accuracy of Object Detection". In:

Zhang, J.-S., Cao, J., and Mao, B. (2017). "Application of deep learning and unmanned aerial vehicle technology in traffic flow monitoring". In: *2017 International Conference on Machine Learning and Cybernetics (ICMLC)*. Vol. 1. IEEE, pp. 189–194.