

Data Safety Prediction Using Bird's Eye View and Social Distancing Monitoring for Penang Roads

Lek Ming Lim¹, Majid Khan Majahar Ali^{1*}, Mohd. Tahir Ismail¹ and Ahmad Sufriil Azlan Mohamed²

¹*School of Mathematical Sciences, Universiti Sains Malaysia, 11800 USM, Penang, Malaysia*

²*School of Computer Sciences, Universiti Sains Malaysia, 11800 USM, Penang, Malaysia*

ABSTRACT

In terms of fatalities, Malaysia ranks third among ASEAN countries. Every year, there is an increase in accidents and fatalities. The state of the road is one factor contributing to near misses. A near miss is an almost-caused accident, an unplanned situation that could result in injury or accidents. The Majlis Bandar Pulau Pinang (MBPP) has installed 1841 closed-circuit television (CCTV) cameras around Penang to monitor traffic and track near miss incidents. When installing CCTVs, the utilisation of video allows resources to be used and optimised in situations when maintaining video memories is difficult and costly. Highways, industrial regions, and city roads are the most typical places where accidents occur. Accidents occurred at 200 per year on average in Penang from 2015 to 2017. Near misses are what create accidents. One of the essential factors in vehicle detection is the "near miss." In this study, You Only Look Once version 3 (YOLOv3) and Faster Region-based Convolutional Neural Network (Faster RCNN) are used to solve transportation issues. In vehicle detection, a faster RCNN was used. Bird's Eye View and Social Distancing Monitoring are used to detect the only vehicle in image processing and observe how near misses occur. This experiment tests different video quality and lengths to compare

test time and error detection percentage. In conclusion, YOLOv3 outperforms Faster RCNN. In high-resolution videos, Faster RCNN outperforms YOLOv3, while in low-resolution videos, YOLOv3 outperforms Faster RCNN.

ARTICLE INFO

Article history:

Received: 29 September 2021

Accepted: 29 December 2021

Published: 15 September 2022

DOI: <https://doi.org/10.47836/pjst.30.4.15>

E-mail addresses:

limlekm@gmail.com (Lek Ming Lim)

majidkhanmajaharali@usm.my (Majid Khan Majahar Ali)

m.tahir@usm.my (Mohd. Tahir Ismail)

sufriil@usm.my (Ahmad Sufriil Azlan Mohamed)

* Corresponding author

Keywords: Bird's eye view, near miss, social distancing monitoring vehicle detection

INTRODUCTION

Road traffic accidents are the leading cause of death among adolescents globally. Road traffic accidents are now the eighth leading cause of death in all age groups worldwide, and they are expected to become the seventh leading cause of death by 2030 (World Health Organization, 2015).

The main concern with the transportation issue is the accuracy of recent (real-time based) and reliable data. Currently, Penang is overly reliant on manually observing data instead of automatically calculating real-time data. The baseline data, POL 37, a police report, is the only manual confidential data that most insurance companies will use to process claims for their clients. However, the data could not be used to determine the road condition because there was insufficient information due to erroneous and missing data.

Majlis Bandar Pulau Pinang (MBPP) already installed 534 Closed-circuit television (CCTV) cameras in 2015 to assist the Council in investigating the condition of road illumination, prohibited dumping activities, and activities on the prohibited ground in the hills. However, the images from all installed cameras are of poor quality. MBPP added 1841 high-resolution CCTV cameras (on both Mainland and Island) to address these concerns in 2019. One of the difficulties discussed concerning CCTV is storage, as MBPP can only preserve a video for 45 days before it is automatically removed. Despite the installation of CCTV cameras, the higher authorities lack an algorithm for calculating and detecting vehicles. Many different types of vehicles complicate the vehicle counting algorithm. No algorithm can be used to implement vehicle detection on a specific road.

Furthermore, POL 37 is a hardcopy police report, which does not assist in the automatic counting of near misses due to the conversion limitations into a visualising report. Rather than relying on hardcopy data, different means must be used. There is no autonomous near miss counting algorithm that can be employed. Due to the lack of research, it is hard to simultaneously calculate near misses from CCTV.

Near miss is one of the issues that the Penang 2030 mission will address to lessen the likelihood of a near miss. To begin, locate near misses on a specific road. Then, because near misses are the causes of accidents, the near misses report and investigation can help to improve road safety. According to the safety triangles or the Heinrich 300-29-1 model, for every 300 near misses, there are 29 minor accidents and one major accident. In addition, accidents raise carbon emissions, which harm the environment. Therefore, the ultimate goal of reporting a near miss is to resolve the incident and take preventative measures to ensure that it does not happen again.

A near miss is an unanticipated event that led to the investigation into the cause of the Malaysian accident. In each of the cases, there was the possibility of an accident occurring, but due to fortunate circumstances, the loss was avoided. According to the researchers, for every significant occurrence, there is a chance of 10 minor incidents and up to 100

near misses (Silva Consultants, 2016). According to Aldred (2016), near miss analyses could reveal information about cyclists' experiences with difficulties related to road user behaviour, culture, and cycling infrastructure. The study also concluded that, based on the experience of near misses, the number of injury incidents could be reduced compared to common types of accidents.

Nevertheless, near misses have sparked deep concern, and they have the potential to be used to investigate factors affecting pedestrian safety. There are numerous crash types in a mixed traffic flow scenario. According to Wang et al. (2020)'s research, low-visibility conditions such as heavy rainy days, foggy days, and nights with insufficient lighting could cause a near miss in traffic flow. A vision-based crash detection framework recognises various objects and crash types from images.

Previous research in Appendix A reveals that there are still some gaps in this study. There are no studies that combine the near miss and vehicle detection applications. Furthermore, most studies employ a survey or questionnaire method to ease data collection. It is due to a lack of accident reports and data. To complete the data, they can only rely on people's experiences. In this study, Social Distancing Monitoring and Bird's Eye View are used in vehicle identification to analyse images and identify near misses.

Vehicle detection is a method for estimating accidents between vehicles and can aid in observing the entire near miss process. Arinaldi (2018) has presented a traffic video system based on the visualisation method that uses important statistics. Vehicle counting, vehicle type, estimated vehicle speed, and vehicle lane change are all included in the statistics. In the previous study, most researchers chose image processing to detect vehicles. Researchers link the model or algorithm method and the software method in their studies. The key reason is that the findings will be visible in a monitoring system. Monitoring systems assist researchers in analysing data from image or video processing. A good strategy to monitor vehicle detection in this study is to use Social Distancing Monitoring and Bird's Eye View methods. Furthermore, these two methods can assist in obtaining accurate and real-time data.

Object detection and vehicle tracking employ a wide range of models and software. CNN, RCNN, Fast RCNN, Faster RCNN, and YOLO are a few examples. However, the models are not used in most near miss studies to estimate the near miss. Zohra et al. (2018) have proposed CNN to improve accuracy and reduce error in the proposed system. Ciberlin et al. (2019) have employed YOLOv3 as object detectors for detection and tracking. Ding and Yang (2019) have demonstrated YOLOv3 to locate parking lots for vehicle detection. The experiment method can improve parking lot detection accuracy while reducing error detection. Huang et al. (2020) have presented traffic flow detection using YOLOv3 and Faster RCNN on 40-second video datasets. Bull et al. (2017) have used Faster RCNN and YOLOv3 to count vehicles in Kuala Lumpur, Malaysia.

In this study, the Social Distancing Monitoring and Bird’s Eye View are used in YOLOv3 to perform image processing on vehicle detection to deal with large and high-resolution images or videos. Near miss accidents can be recorded and observed using image processing and vehicle detection. These analyses can forecast future near miss accidents and pinpoint the source of the problems.

MATERIALS AND METHODS

Convolutional Neural Network

Convolutional Neural Networks (CNN) are highly popular in the deep learning community. These CNN models are used in various applications, the most common of which are image and video processing projects (Rawat & Wang, 2017). A CNN is a computational model that also employs several layers of neurons and is composed of one or more convolutional layers that can be completely linked or pooled. Furthermore, these convolutional layers provide feature maps that record an area of the picture, divided into rectangles and sent out for non-linear processing (Qin et al., 2018).

Figure 1 shows the operation of a convolutional neural network. First, provide the input image. Next, put the image into the convolution layer. Then, perform pooling to reduce the dimensionality. Finally, examine the outcomes to see if they meet expectations. Otherwise, return to the convolution layer and display the results.

According to Behl et al. (2014), convolution is a mathematical operation performed on two functions (f and g) to generate a third function defined as ($f * g$) that explains how the form of one is changed by the other in Equation 1.

$$(f * g)(t) := \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau = \int_{-\infty}^{\infty} f(t - \tau)g(\tau)d\tau \tag{1}$$

Although the symbol t as in Equation 1, does not always indicate the time domain. The convolution formula can be written as a weighted average of the function $f(\tau)$ at time t , where $g(-\tau)$ represents the weight, and just the amount t is shifted. The weighting function emphasises certain aspects of the input function as t changes. As a result, the integration

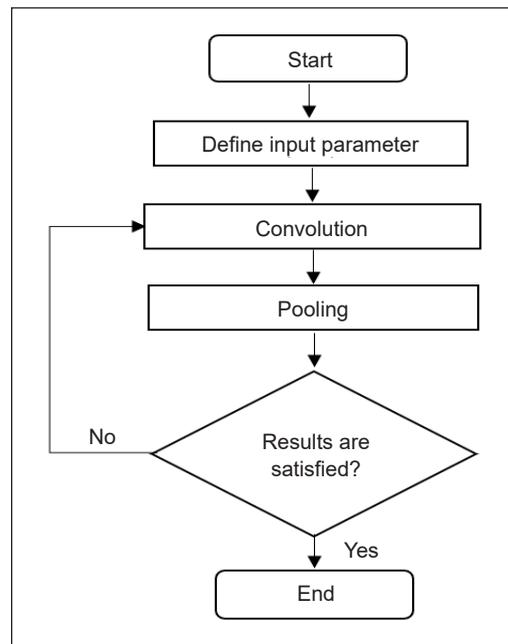


Figure 1. CNN flow chart

limitations can be shortened. Functions that are only supported on (for example, 0 for negative arguments), resulting in Equation 2:

$$(f * g)(t) := \int_0^t f(\tau)g(t - \tau)d\tau \tag{2}$$

Convolutional neural networks employ many cascaded convolution kernels in machine vision and artificial intelligence applications. CNN models have a consistent structure that consists of alternating convolutional layers and pooling layers (typically, each pooling layer is arranged after a convolutional layer), which serve as feature extraction.

According to Albelwi and Mahmood (2017), the last layers consist of a limited number of fully connected layers, with the final layer being a softmax classifier used for image classification. Over K classes, the softmax classifier estimates the posterior probability of each class label.

$$y_i = \frac{\exp(-z_i)}{\sum_{j=i}^K \exp(z_j)} \tag{3}$$

Where z_i is any real number and reflects the softmax function's input values, the bottom part of Equation 3 is the normalisation terms, required to ensure that all the function's output values add up to i .

The term "hyperparameters" refers to numerous settings that influence learning. For example, CNN employs more hyperparameters than a typical multilayer perceptron (MLP). In the CNN architecture design, the algorithm for the CNN is described by a structural hyperparameter λ that encompasses the CNN architecture design is shown as in Equation 4:

$$\lambda = \left((\lambda_1^i, \lambda_2^i, \lambda_3^i, \lambda_4^i)_{i=1, M_C}, \dots, (\lambda_j^i)_{j=1, N_f} \right) \tag{4}$$

where $\lambda \in \Psi$ determines the domain for each hyperparameter, M_C is the number of convolutional layers, N_f is the number of fully connected λ_1^i is the number of filters, λ_2^i is the filter size, λ_3^i is the pooling locations and size, and λ_4^i is the stride step.

Social Distancing Monitoring

Social distancing monitoring is a technique for measuring the distance between vehicles and predicting the likelihood of an accident occurring. Calculate the Euclidean distance between all detected boxes and filter out or flag vehicles close to each other, indicating that the vehicles are at risk by using this method.

Figure 2 shows the flowchart for Social Distancing Monitoring. Input the video into the programme as the first step. Then, use object detection to find the single vehicle in the image or frame. The pairwise distances between all detected vehicles should then

be computed. Then, using the horizontal and vertical unit lengths, calculate the distance between the two vehicles. The unit length is measured in centimetres (cm). If the distance is greater than 180 cm, green boxes would appear (safe). Also, the yellow boxes (near miss) are displayed when the distance is between 50 cm and 180 cm; otherwise, red boxes (high risk) are displayed when a distance is less than 50 cm.

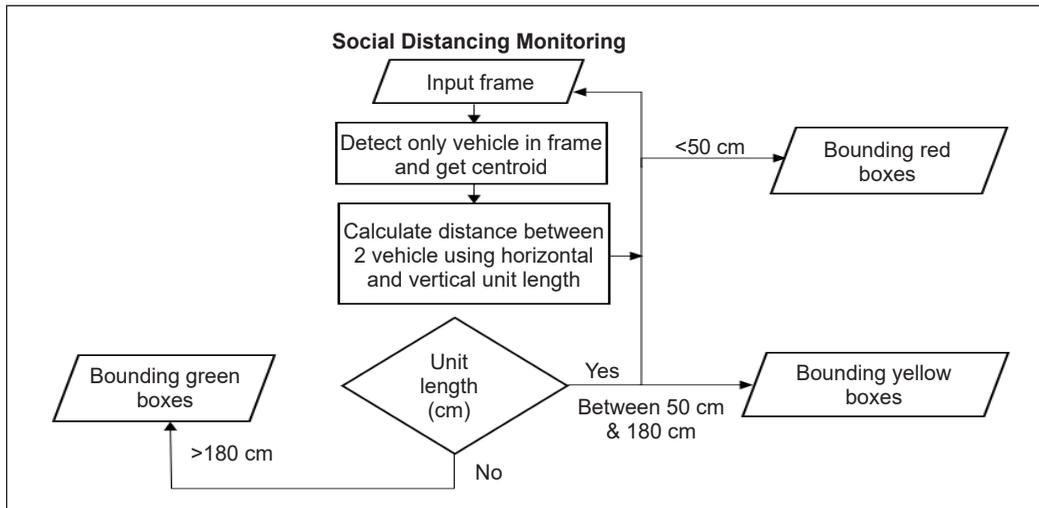


Figure 2. Social distancing monitoring algorithm

Bird's Eye View

The Bird's Eye View method calculates the near miss rate by displaying the distances between points (vehicles) in a specific frame box. The near miss is calculated based on the distance between the vehicles. When the distance between two vehicles is very close to causing an accident, it is called a near miss. The near miss distance between vehicles estimation and how close the threshold for near miss distance between vehicles estimation must be implemented.

Figure 3 shows the flowchart using the Bird's Eye View method. Input the video into the programme as the first step. Then, using object detection, find the only vehicle in the image or frame and calculate the centroid. The project was then discovered from a Bird's Eye View. Then, the horizontal and vertical unit lengths compute the distance between two vehicles. The length of a unit is measured in centimetres (cm). Green dots appear (safe) if the distance is greater than 180 cm. The yellow dots (near miss) appear if the distance is between 50 cm and 180 cm. Otherwise, the red dots (high risk) appear at less than 50 cm.

Figure 4 shows how the ImageJ application can measure the distance between each pixel in the image in real life. In this video, 209.1220 pixels equal 32 rocks, which equals 96 feet, or 2926.08 cm. Hence, the image displays 0.0715 pixel = 1 cm or 0.0715 pixel/cm with a perspective angle ranging from 10 to 15. As a result, the two points represent

the threshold of a near miss between two vehicles. When the two blue points are fixed, the number of pixels fluctuates depending on their width. For example, 12.87 pixels = 180 cm (constant distance).

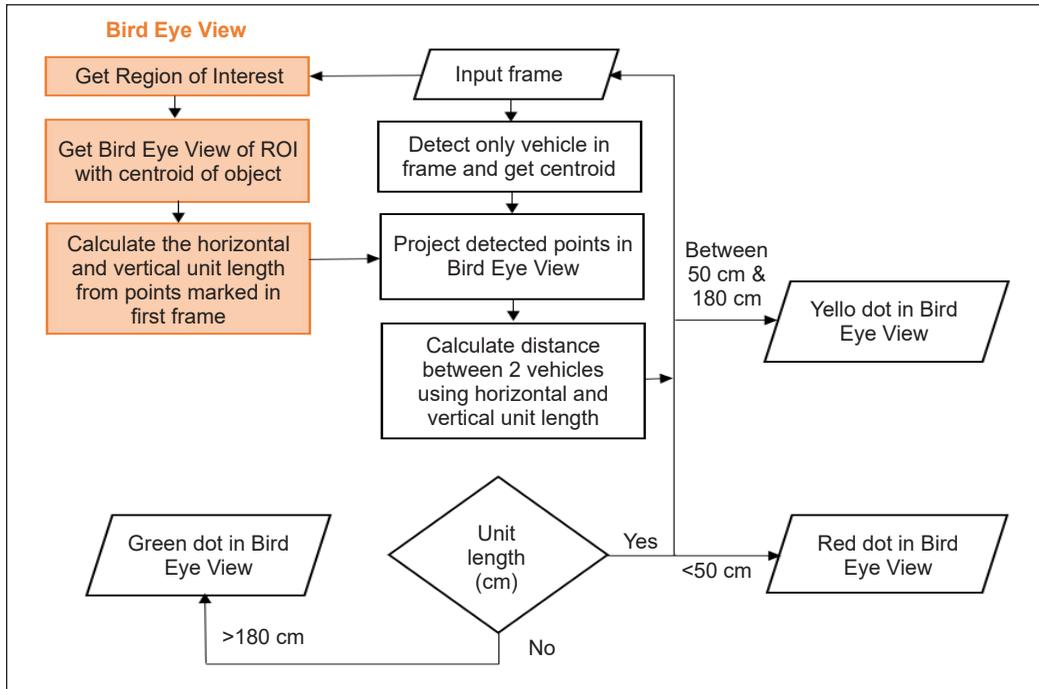


Figure 3. Bird's Eye View algorithm

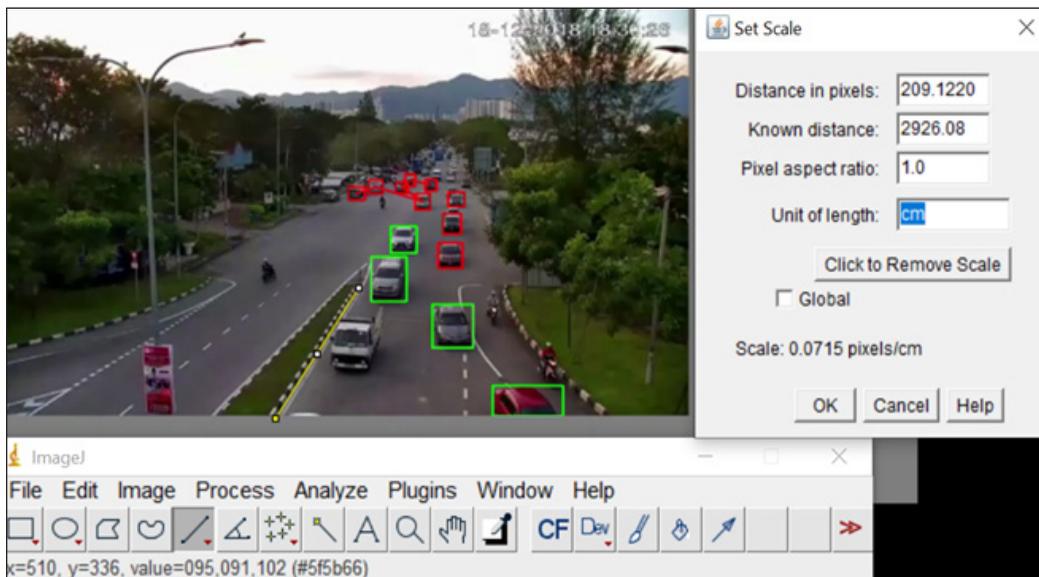


Figure 4. Calculation of unit length in the Tun Dr Lim Chong Eu Expressway video

the threshold of a near miss between two vehicles. When the two blue points are fixed, the number of pixels fluctuates depending on their width. For example, 12.87 pixels = 180 cm (constant distance).

As shown in Figure 5, the ImageJ application measured the distance between pixels in an image in real life (Rahman et al., 2012). In this video, 134.1641 pixels = 12 rocks = 36 feet = 1097.28 cm. As a consequence, the image displays 0.1223 pixels = 1 cm or 0.1223 pixels/cm with a perspective angle of 10 to 15 degrees. For example, 22.014 pixels = 180 cm (constant distance).

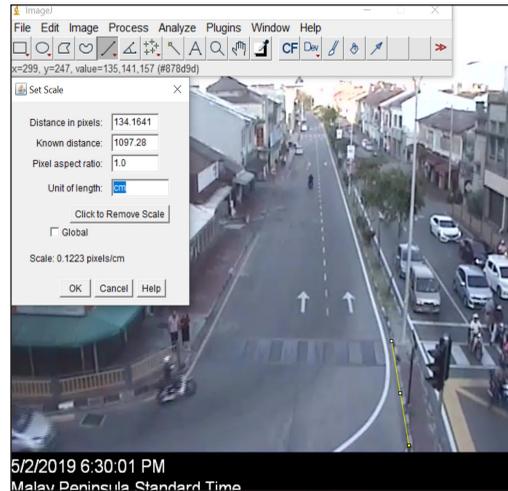


Figure 5. Calculation of unit length in the Lim Chwee Leong Road video

Faster RCNN

Convolutional neural networks (CNN) are mostly used for image classification, whereas RCNN is typically used for object detection. Faster RCNN was developed in June 2015. The selective search is replaced by a region proposal network (RPN) and a detector based on Fast RCNN for more accurate object detection while reducing the number of region proposals.

Faster RCNN is focused on and used to detect vehicles in this study. It is a deep convolutional neural network that detects and classifies objects in images. RPN and Fast RCNN are the two modules that makeup Faster RCNN. RPN generates region suggestions, while Fast RCNN is used to identify objects in the suggested regions. It employs the concept of attention in neural networks via the RPN function, and thus RPN guides the Fast RCNN as a detector to locate objects in images (Gad, 2020).

The datasets are derived from the videos. First, the video is divided into frames. Then label the vehicles in the frames. Next, train the model with the labelled file until the loss functions are less than 0.5. The training frames must account for 40% of the total frames and 60% of the testing frames for detection. Finally, the Faster RCNN is used in the experiment.

RESULTS AND DISCUSSION

YOLOv3 Results

The video Social Distancing Monitoring monitors the distance between vehicles and predicts the likelihood of accidents occurring (Vinitha & Velantina, 2020). In addition, a bird's eye view is used to show the distances between points (vehicles) in a specific frame box and estimate that near misses occur. These two methods are used in this study

to demonstrate how near misses occur and to estimate the distance between vehicles in video and real life.

YOLOv3 is used to detect vehicles for the Social Distancing Monitoring and Bird's Eye View (Ong, 2020). Other datasets, such as motorcycles, people, and trucks, are being filtered out, leaving only cars to be detected. In the video, the results showed red bounding boxes (high-risk accident), yellow bounding boxes (near miss detected), and green bounding boxes (no accident occurred).

Social Distancing Monitoring detects all vehicles in the video, whereas Bird's Eye View uses Region of Interest (ROI) to detect vehicles. Furthermore, the distance settings in Social Distancing and Bird's Eye View are identical. As a result, the outcomes of both methods are the same. The locations of the dots in Bird's Eye View, for example, can be shown in the detected vehicles in Social Distancing Monitoring. Figure 6 depicts the ROI points on Lebuhraya Tun Dr Lim Chong Eu.

Figure 7 shows only vehicle detection using YOLOv3 to detect the likelihood of collisions and near miss between vehicles. Result Tun Dr Lim Chong Eu Expressway videos were shot between 18/12/2018, 6:30:15p.m. and 18/12/2018, 6:31:34p.m. Only cars within the specified rectangle zone would be displayed in the bird's eye view. Due to the poor viewpoint, results outside the rectangular zone are ignored, and only cars within the ROI are recorded.



Figure 6. The ROI for Tun Dr Lim Chong Eu Expressway and the threshold ratio

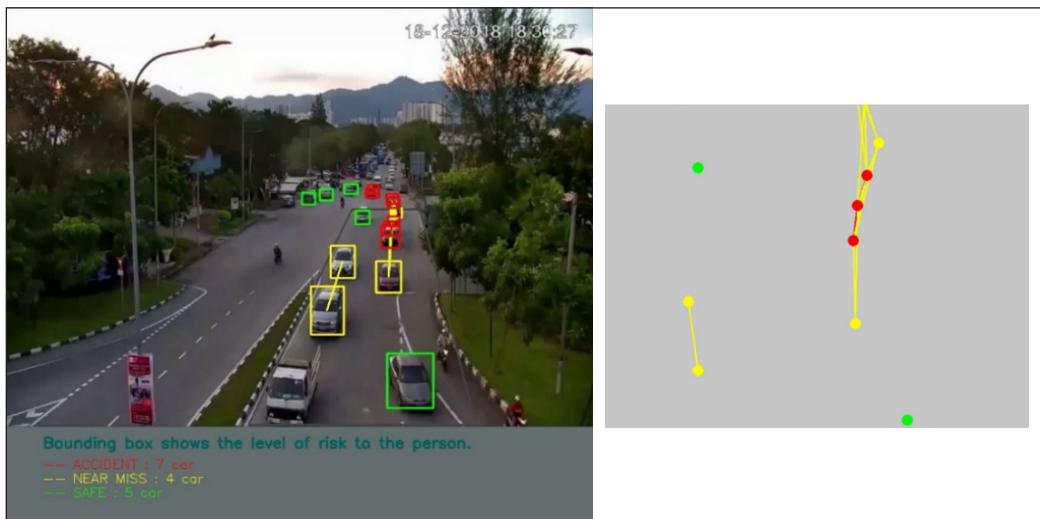


Figure 7. Tun Dr Lim Chong Eu Expressway near miss detection result

Table 1 compares different lengths of the same video in the 20s, 40s, 60s, and 80s of high-quality videos in YOLOv3. Three videos with a 10-minute duration and 30 frames per second were taken from YouTube to conduct the experiments in Sonnleitner et al. (2020)'s study. The running time increases as the length of the video increases. It is one of the reasons why a 20-second video was chosen for this study instead of an 80-second video because the computational time for the image process using YOLOv3 is too long for the computational process. Although the data in the 80-second video is more reliable than the data in the 20-second video, manually counting data is a major task for data recording—the total number of frames taken increases as the video length increases.

As the duration of videos in image processing increases, error detection, near miss detection, and accident detection can be observed and recorded in greater detail. Motorcycles, trucks, and lorries are recognised as vehicles in error detection. The percentage of near miss detection and accident detection in the 20s video is higher than in the 80s video. It demonstrates that the vehicles in the first 20 seconds of the video are too near to each other when stopping in front of a traffic light or when there is congestion. After the 20s, the vehicles began to move and maintain a safe distance from one another. As a result, near miss and accident detection percentages were reduced from 100% to 84.61% and 33.33% to 28.44%.

Table 1
Comparison of high-quality videos in YOLOv3

CCTV time	6:30:15 p.m. – 6:30:34 p.m.	6:30:15 p.m. – 6:30:54 p.m.	6:30:15 p.m. – 6:31:14 p.m.	6:30:15 p.m. – 6:31:34 p.m.	
Length of video	20s	40s	60s	80s	
Computational time	445 seconds	838 seconds	1655 seconds	2501 seconds	
Total number of frames	600	1198	1798	2398	
Error detection	Number of frames	10	106	120	226
	Percentage	$10/600 \times 100\% =$ 1.67%	$106/1198 \times 100\%$ = 8.85%	$120/1798 \times 100\%$ = 6.68%	$226/2398 \times 100\%$ = 9.42%
	Object detected	Motorcycle	Motorcycle, truck	Motorcycle, truck	Motorcycle, truck, lorry
Near Miss detection	Number of frames	600	1153	1734	2029
	Percentage	$600/600 \times 100\%$ = 100%	$1153/1198 \times$ $100\% =$ 96.24%	$1734/1798 \times$ $100\% =$ 96.44%	$2029/2398 \times$ $100\% =$ 84.61%
Accident detection	Number of frames	200	392	385	682
	Percentage	$200/600 \times 100\%$ = 33.33%	$392/1198 \times 100\%$ = 32.72%	$385/1798 \times 100\%$ = 21.41%	$682/2398 \times 100\%$ = 28.44%

Figure 8 shows the ROI points on Lim Chwee Leong Road. In contrast, Figure 9 shows only vehicle detection using YOLOv3 to detect the likelihood of collisions and near miss between vehicles. Result Lim Chwee Leong Road videos were shot between 5/2/2019, 6:30:15p.m. and 5/2/2019, 6:31:34p.m. The Bird's Eye View only shows vehicles within the specified rectangular area. Results outside the rectangular area are ignored as the viewpoint is weak and only cars within the ROI are counted.

Table 2 compares different lengths of the same video, which are in the 20s, 40s, 60s, and 80s of low-quality videos in YOLOv3. The computing time increases as the length of the video increases. It is one of the reasons why a 20-second video was chosen for this study instead of an 80-second video because the computational time for the image process using YOLOv3 is too long for the computational process. Therefore, in this study, image processing was carried out using videos with a length of 80 seconds, which is more trustworthy than videos of 20 seconds. However, the problem of counting data in detection was manually counted, which took a long time and was a significant advancement—the total number of frames taken increases with the length of the video.

As the duration of videos processed in image processing increases, error detection, near miss detection, and accident detection can be observed and recorded in greater detail. Motorcycles, tricycles, and buses are identified as vehicles in error detection. From the 20s video to the 80s video, the percentage of near miss detection and accident detection decreased from 37.63% to 27.23% and 1.17% to 0.5%, respectively.



Figure 8. Result Lim Chwee Leong Road ROI and set threshold ratio

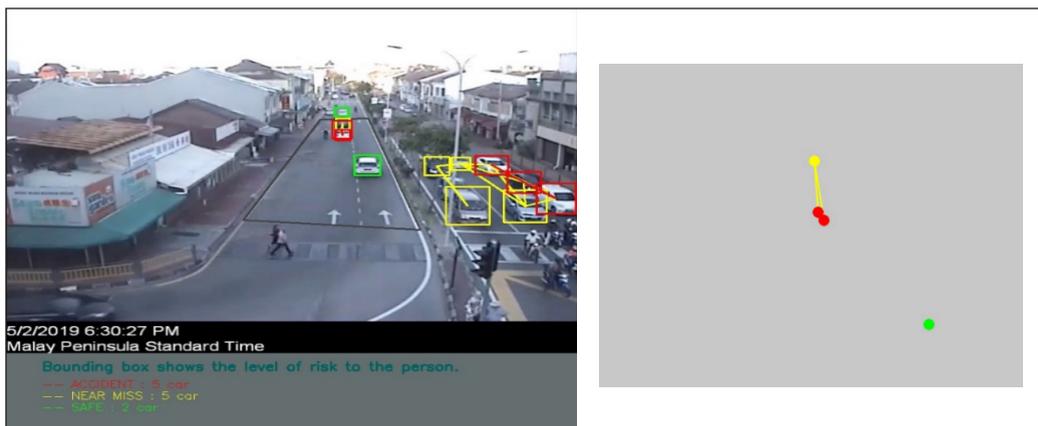


Figure 9. Result Lim Chwee Leong Road near miss detection

In this low-quality video (Lim Chwee Leong Road), near miss and accident probability is lower than in Tun Dr Lim Chong Eu Expressway's video.

Table 2
Comparison of low-quality videos in YOLOv3

CCTV time	6:30:15 p.m.– 6:30:34 p.m.	6:30:15 p.m.– 6:30:54 p.m.	6:30:15 p.m.– 6:31:14 p.m.	6:30:15 p.m.– 6:31:34 p.m.	
Length of video	20s	40s	60s	80s	
Computational time	572 seconds	1255 seconds	2083 seconds	2680 seconds	
Total number of frames	598	1198	1798	2398	
Error detection	Number of frames	2	120	154	
	Percentage	$2/598 \times 100\% =$ 0.33%	$120/1198 \times 100\% =$ 10.02%	$154/1798 \times 100\% =$ 8.57%	$154/2398 \times 100\% =$ 6.42%
	Object detected	Motorcycle	Motorcycles, tricycle, bus	Motorcycles, tricycle, bus	Motorcycles, tricycle, bus
Near Miss detection	Number of frame detection	225	406	438	509
	Percentage	$225/598 \times 100\% =$ 37.63%	$406/1198 \times 100\% =$ 33.89%	$438/1798 \times 100\% =$ 24.36%	$509/2398 \times 100\% =$ 21.23%
Accident detection	Number of frame detection	7	11	12	12
	Percentage	$7/598 \times 100\% =$ 1.17%	$11/1198 \times 100\% =$ 0.92%	$12/1798 \times 100\% =$ 0.66%	$12/2398 \times 100\% =$ 0.5%

Huang et al. (2020) collected data from three videos shot in the 1940s for their paper. The videos show various weather and scenario scenarios. For video traffic monitoring, the YOLOv3 algorithm is used. They also mention the location of the video collection and the field of vision. If the video collection is too large, this will result in missed detection. Error detection occurs when the field of view is too large. The experiment used video lengths from the 20s, 40s, 60s, and 80s in this study. Weather, other scenarios, and camera location could all play a role in future work.

Furthermore, Cepni et al. (2020) presented video collected from UAVs (crewless aerial vehicles) with a resolution of 1280x720 and video collected at a terrestrial quality of 1080x1920, which was tested in the experiment. The videos are 24 fps and a one-minute-long video. When comparing both videos, terrestrial videos outperform model accuracy and estimation. Therefore, two videos are tested in this study: high-quality (Tun Dr Lim Chong Eu Expressway) and low-quality videos (Lim Chwee Leong Road).

Figure 10 shows the test time in high-quality and low-quality videos. Low-quality videos require more computational time than high-quality videos.

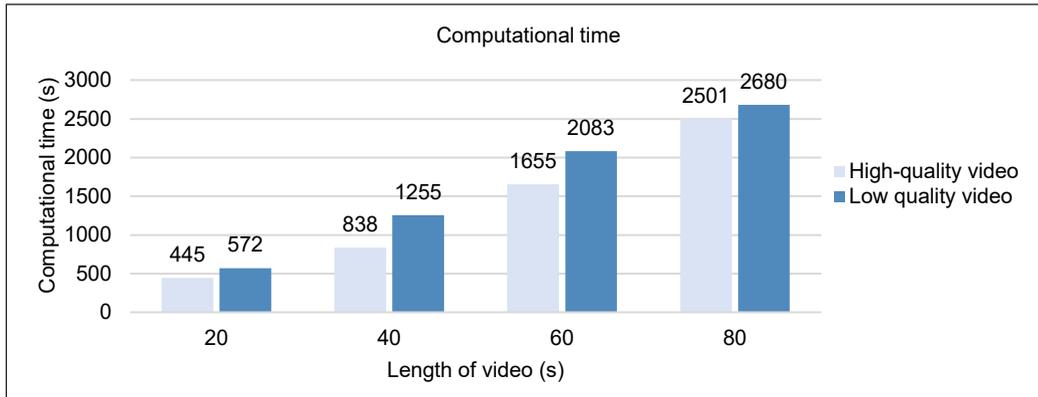


Figure 10. Comparison of computational time between high-quality videos and low-quality videos in YOLOv3

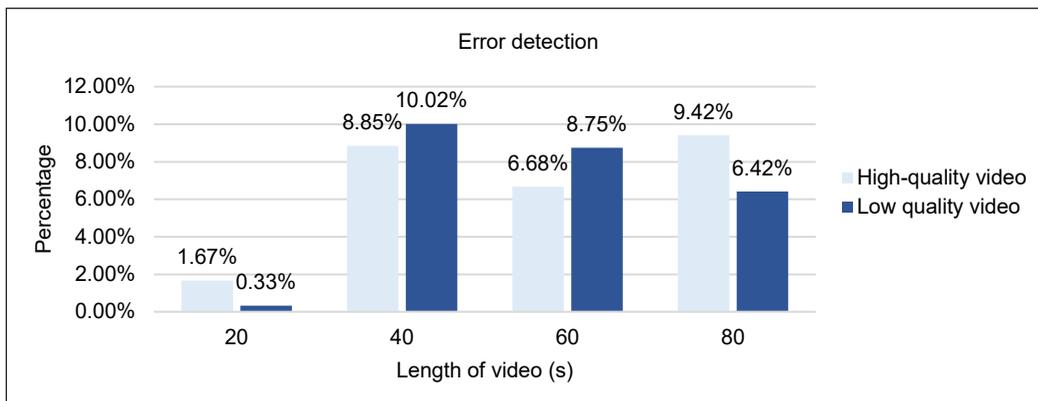


Figure 11. Comparison of error detection between high-quality videos and low-quality videos in YOLOv3

Figure 11 represents error detection in high-quality and low-quality videos. In YOLOv3, error detection occurred, such as detecting a motorcycle, a truck, and a lorry in high-quality videos and detecting motorcycles, a tricycle, and a bus as the car. A proportion of error analysis for YOLO and Fast RCNN is presented in Dixit et al.'s (2019) paper. Compared to Fast-RCNN, most of the errors in YOLO are localisation errors.

Figure 12 presents the detection of errors in high-quality and low-quality videos. This analysis supports the theory that near miss occurs during the black spot's peak hour. Aside from driver behaviour, the root cause of near misses is an enormous issue. According to Matsui et al.'s (2013) research, the development of driving safety devices necessitates detailed functions of the contact scene between the car and the pedestrian to reduce the number of fatalities and the severity of injuries in Japan. However, due to a lack of data from real-world accidents, the researchers focused on near miss situations. As a result, in the video taken from the black spot area, Social Distancing Monitoring and Bird's Eye View are used to detect the near miss.

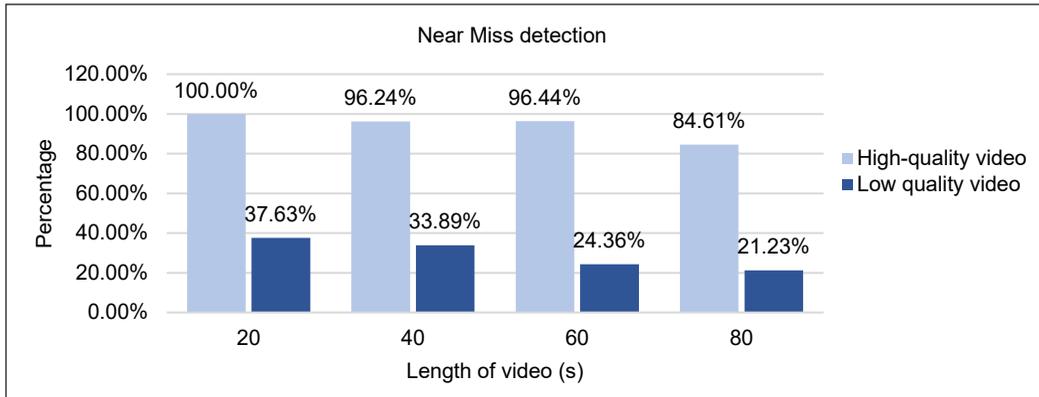


Figure 12. Comparison of near miss detection between high-quality videos and low-quality videos in YOLOv3

According to Vinita and Velantina’s (2020) research, social distance detection tools can monitor the public to keep themselves safe from other people using video from surveillance cameras. By combining Social Distancing Monitoring and Bird’s Eye View, the experiment sheds new light on vehicle detection. The methods described above also detect vehicles at a safe distance to avoid near misses or accidents and do not exceed the threshold ratio set in the algorithm.

The videos provided by MBPP vary in quality depending on location. They are not in the same location and have different video quality. In the introduction, MBPP installed 534 CCTV cameras in 2015 and 1841 CCTV cameras in 2019. Some locations installed CCTV cameras in 2015, but some places are not good quality compared to good quality places.

Faster RCNN Results

Faster RCNN also detected vehicles on Tun Dr Lim Chong Eu Expressway and Lim Chwee Leong Road. The videos on Tun Dr Lim Chong Eu Expressway are high quality, whereas the videos on Lim Chwee Leong Road are low quality. Both videos are tested in various lengths, including the 20s, 40s, 60s, and 80s.

Figure 13 captures the detection of a vehicle on Tun Dr Lim Chong Eu Expressway. Fast RCNN detects only cars in the video and uses green bounding boxes to show the percentage similarity of objects in the dataset.

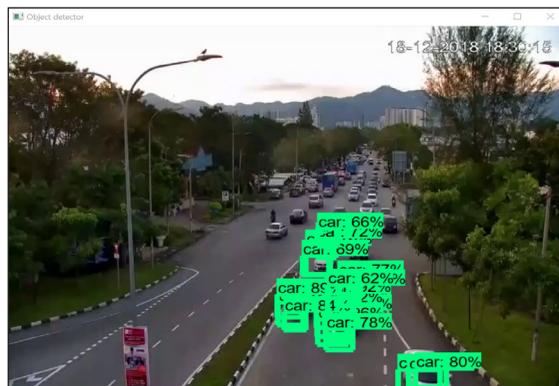


Figure 13. Result of Tun Dr Lim Chong Eu Expressway vehicle detection

Table 3 compares the lengths of the video Tun Dr Lim Chong Eu

Expressway, covering the 20s, 40s, 60s, and 80s of high-quality videos in Faster RCNN. The computational time required increases in proportion to the length of the video. Therefore, compared to an 80-second video, this study chose a 20-second video duration because the computational time is too long for the vehicle detection process when using Faster RCNN. The total number of frames taken increases with the length of the video. The dataset from the video frames is used to train a Faster RCNN. As a result, when compared to YOLO, the computational time is lengthy (Alganci et al., 2020). In this video, Faster RCNN outperforms YOLO compares the accuracy and speed of the Faster RCNN, SSD, YOLO, and NVIDIA (Dixit et al., 2019). Faster-RCNN is the most accurate algorithm with the slowest speed, whereas YOLO is super-fast with low accuracy. In this study, the number of frame errors in detection is less than 1% for four videos of varying lengths. The object that caused the error is a banner.

Table 3
Comparison of high-quality videos in Faster RCNN

CCTV time	6:30:15 p.m. – 6:30:34 p.m.	6:30:15 p.m. – 6:30:54 p.m.	6:30:15 p.m. – 6:31:14 p.m.	6:30:15 p.m. – 6:31:34 p.m.	
Length of video	20s	40s	60s	80s	
Computational time	747 seconds	1496 seconds	2272 seconds	3111 seconds	
Total number of frames	599	1200	1798	2399	
Error detection	Number of frames	3	3	3	
	Percentage	3/599 x 100% = 0.5 %	3/1200 x 100% = 0.25%	3/1798 x 100% = 0.17%	3/2399 x 100% = 0.12%
	Object detected	Banner	Banner	Banner	Banner

Figure 14 captures the outcome of the Lim Chwee Leong Road vehicle detection. Fast RCNN detects only vehicles in the video and uses green colour bounding boxes to show the percentage similarity of objects in the dataset. When executing the Faster RCNN, the error detection is proven in the video.

Table 4 compares the lengths of the video Tun Dr Lim Chong Eu Expressway, covering the 20s, 40s, 60s, and 80s of low-quality videos in Faster RCNN. When the video's duration is longer, the amount of computing time required increases. Compared to an 80-second video, this study chose a 20-second video



Figure 14. Result Lim Chwee Leong Road vehicle detection

duration because the computational time for the vehicle detection using Faster RCNN is too long for the computational process. The total number of frames taken increases as the video length increases. Faster RCNN is trained using the video frames as a dataset. Therefore, compared to YOLO, the computational time is lengthy (Srivastava et al., 2021). For the four videos of varying lengths, the number of frame errors in detection is 100%. The roof, unknown objects, and the bus are error-detected objects. Due to the low-quality video (Aqqa et al., 2019) and small dataset (Cao et al., 2019) used to train Faster RCNN, it cannot detect vehicles as accurately as YOLO in Table 4.

Table 4
Comparison of low-quality videos in Faster RCNN

CCTV time	6:30:15 p.m. – 6:30:34 p.m.	6:30:15 p.m. – 6:30:54 p.m.	6:30:15 p.m. – 6:31:14 p.m.	6:30:15 p.m. – 6:31:34 p.m.
Length of video	20s	40s	60s	80s
Computational time	757 seconds	1642 seconds	2449 seconds	3218 seconds
Total number of frames	599	1200	1798	2398
Number of frames	599	1200	1798	2398
Percentage	$599/599 \times 100\%$ = 100%	$1200/1200 \times$ $100\% = \mathbf{100\%}$	$1798/1798 \times$ $100\% = \mathbf{100\%}$	$2398/2398 \times$ $100\% = \mathbf{100\%}$
Object detected	Roof, unknown object	Roof, unknown object, bus	Roof, unknown object, bus	Roof, unknown object, bus

According to Cao et al. (2019), the researchers proposed an improved Faster RCNN and used the TT100K (Tsinghua-Tencent 100K) dataset, which saves 100,000 images, including 30,000 traffic-sign occurrences. Then, compare it to other research papers that also use Faster RCNN. However, only 400 video frame samples were labelled and trained for this study. Faster RCNN requires more sample data to train the algorithm.

The KITTI dataset trains the Faster RCNN in Zhang et al.'s (2018) research. The test video was shot on a real-life road in 1280x720 resolution. The dataset for this study was derived from CCTV video, including the training and labelling images. Following that, the images from the experiment are used to train the model.

Figure 15 represents the test time in high-quality and low-quality videos. Low-quality videos require more computational time than high-quality videos.

Figure 16 shows the detection of errors in high-quality and low-quality videos. According to Aqqa et al. (2019), video quality is an important factor, often overlooked. The video is tested using Faster RCNN, SSD, YOLO, and RetinaNet for object detection at various video compression levels to investigate the quality distortion caused by compression artefacts during video capture. In this study, low-quality video is tested using YOLOv3 and Faster RCNN. There are numerous errors found in the results.

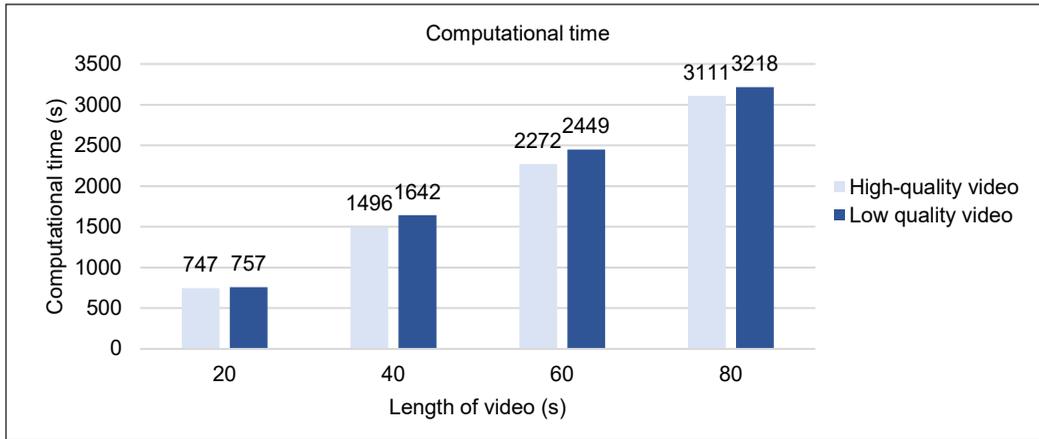


Figure 15. Comparison of computational time between high-quality videos and low-quality videos in Faster RCNN

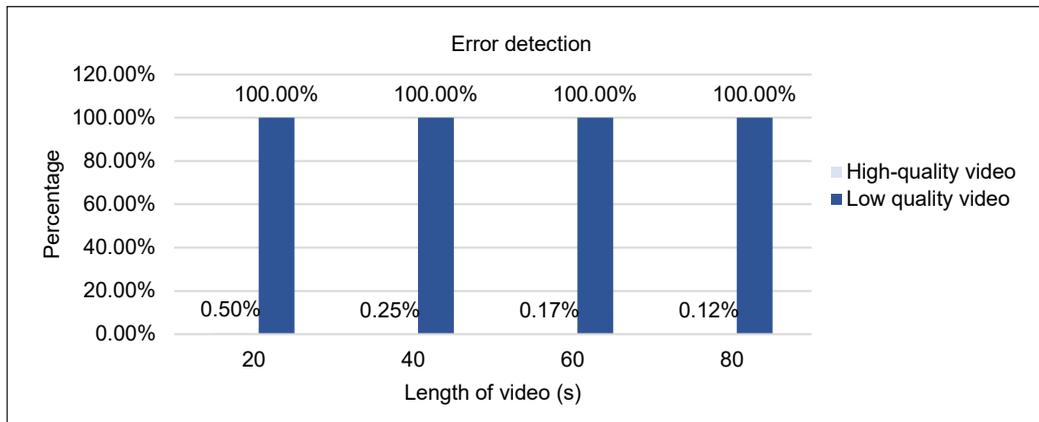


Figure 16. Comparison of error detection between high and low-quality videos in Faster RCNN

Comparison Between YOLOv3 and Faster RCNN

High-quality (Tun Dr Lim Chong Eu Expressway) and low-quality (Lim Chwee Leong Road) videos were tested in YOLOv3 and Faster RCNN. The similarities between YOLOv3 and Faster RCNN are that they use anchor boxes based on network structure, bounding boxes, and the same length of videos for the experiment. However, there are also differences between YOLOv3 and Faster RCNN. Table 5 compares YOLOv3 and Faster RCNN in high-quality videos. The following comparison is based on Tables 1 and 3.

The video provided by the Penang state government impacts the reliability of this data. However, the video contains reliable information that can be saved as historical data for future use. According to Calles et al. (2017), near miss cases for transportation provide a good opportunity to determine whether there are problems and intervene before the actual accident. Behaviour and driving experience are two factors that can explain the risk of near

Table 5
Comparison in high-quality videos

YOLOv3	Difference	Faster RCNN
Social Distancing Monitoring and Bird's Eye View	Method	Vehicle detection
Fast	Speed	Slow
Higher than Faster RCNN	Error detection	Low
Very high	Near Miss detection	null
Low	Accident detection	null

Table 6
Comparison in low-quality videos

YOLOv3	Difference	Faster RCNN
Social Distancing Monitoring and Bird's Eye View	Method	Vehicle detection
Fast	Speed	Slow
Low	Error detection	Very high
High	Near Miss detection	null
Very low	Accident detection	null

miss accidents among young drivers. Collaboration with hospitals and insurance companies should be considered in future studies to obtain more complete data and reduce data flaws.

Table 6 shows a comparison in low-quality videos of YOLOv3 and Faster RCNN. The following comparison is based on Tables 2 and 4.

According to Alganci et al. (2020), the researchers concluded that YOLOv3 has a shorter processing time. In this study, YOLOv3 takes less computational time than Faster RCNN in high-quality and low-quality videos.

Tables 5 and 6 show that the Faster RCNN lacks near miss and accident detection data. In addition, Social Distancing Monitoring and Bird's Eye View techniques are not available at Faster RCNN.

Finally, it is possible to compare YOLOv3 and Faster RCNN, where Faster RCNN requires more computational time than YOLOv3 in the high-quality and low-quality videos. Since YOLOv3 is a simpler architecture, Faster RCNN is trained to perform classification and bounding box regression simultaneously.

In the percentage of error detection comparison, Faster RCNN exhibited more accurate data than YOLOv3 in the videos of Tun Dr Lim Chong Eu Expressway but not in the videos of Lim Chwee Leong Road. Faster RCNN required more dataset samples based on the videos to train the algorithm, whereas YOLOv3 did not require any training. After all, its dataset already trained it. In vehicle detection, both algorithms show detection errors. YOLOv3 detected motorcycles, trucks, and lorries in Tun Dr Lim Chong Eu Expressway videos. The videos of Lim Chwee Leong Road detected motorcycles, tricycles, and buses as vehicles. In Tun Dr Lim Chong Eu Expressway videos, faster RCNN detected banners

as vehicles, while in Lim Chwee Leong Road, it detected roofs, unknown objects, and a bus as vehicles.

CONCLUSION

This research aims to investigate near miss cases in Pulau Pinang to reduce the number of accidents and carbon emissions in the city. This study has the potential to achieve the goals and SDGs outlined in the Penang 2030 mission. It is necessary to identify the black spot area in Tun Dr Lim Chong Eu Expressway and select the peak hours (18/12/2018, 6:30:15p.m. to 18/12/2018, 6:31:34p.m.) with a high-quality video and Lim Chwee Leong Road and the peak hours (5/2/2019, 6:30:15p.m. to 5/2/2019, 6:31:34p.m.) with low-quality video to conduct the vehicle detection by using YOLOv3 (Social Distancing Monitoring and Bird's Eye View method) and Faster RCNN. Faster RCNN takes longer than YOLOv3 computational time due to vehicle detection. In the high-quality video, Faster RCNN produced more accurate data than YOLOv3, while YOLOv3 produced more accurate data in the low-quality video. In YOLOv3, a near miss is likely higher on Tun Dr Lim Chong Eu Expressway than on Lim Chwee Leong Road. Image or video quality needs to be improved in future work. An important criterion is the camera angle. If the calculation can be converted into an autonomous calculation, the duration of the videos can be extended. Near miss is predicted to count automatically. It is expected that the current event will be used to forecast future events such as seasonal changes and the achievement of the Penang 2030 mission.

ACKNOWLEDGEMENT

We would like to thank the MBPP CCTV Team that assisted by providing necessary information and School of Mathematical Sciences, Universiti Science Malaysia for the funding.

REFERENCES

- Albelwi, S., & Mahmood, A. (2017). A framework for designing the architectures of deep convolutional neural networks. *Entropy*, 19 (242), 1-20. <https://doi.org/10.3390/e19060242>
- Aldred, R. (2016). Cycling near misses: Their frequency, impact and prevention. *Transport Research Part A*, 90(1), 69-83. <https://doi.org/10.1016/j.tra.2016.04.016>
- Aldred, R., & Crosweller, S. (2015). Investigating the rates and impacts of near misses and related incidents among UK cyclists. *Journal of Transport & Health*, 2(3), 379-393. <https://doi.org/10.1016/j.jth.2015.05.006>
- Alganci, U., Soydas, M., & Sertel, E. (2020). Comparative research on deep learning approaches for airplane detection from very high-resolution satellite images. *Remote sensing*, 12(3), Article No. 458. <https://doi.org/10.3390/rs12030458>

- Aqqa, M., Mantini, P., & Shah, S. K. (2019, February). Understanding how video quality affects object detection algorithms. In *VISIGRAPP (5: VISAPP)* (pp. 96-104). Science and Technology Publications. <https://doi.org/10.5220/0007401600960104>
- Arinaldi, A., Pradana, J., A., & Gurusniaga, A., A. (2018). Detection and classification of vehicles for traffic video analytics. *Procedia Computer Science*, *144*, 259-268. <https://doi.org/10.1016/j.procs.2018.10.527>
- Behl, A., Bhatia, A., & Puri, A. (2014). Convolution and applications of convolution. *International Journal of Innovative Research in Technology (IJIRT)*, *1*(6), 2123-2126.
- Bull, C., B., Hagen, L., A., V., Lubin, A., Shivaraman, G., & Chibbaro, D. (2017). *Predictable is preventable: Tracking pedestrian near-miss incidents*. New Jersey Safe Routes to School Resource Center & Alan M. Voorhees Transportation Center. <https://www.njcrossingguards.org/wp-content/uploads/2017/11/Ped-Near-Miss-report-5.25.17.pdf>
- Calles, M., B., Nelson, T., & Winters, M. (2017). Comparing crowdsourced near-miss and collision cycling data and official bike safety reporting. *Transportation Research Record: Journal of the Transportation Research Board*, *2662*(1), 1-11. <https://doi.org/10.3141/2662-01>
- Cao, C., Wang, B., Zhang, W., Zeng, X., Yan, X., Feng, Z., Liu, Y., & Wu, Z. (2019). An improved faster R-CNN for small object detection. *IEEE Access*, *7*, 106838-106846. <https://doi.org/10.1109/ACCESS.2019.2932731>
- Cepni, S., Atik, M. E., & Druan, Z. (2020). Vehicle detection using different deep learning algorithms from image sequence. *Baltic Journal of Modern Computing*, *8*(2), 347-358. <https://doi.org/10.22364/bjmc.2020.8.2.10>
- Ciberlin, J., Grbic, R., Teslic, N., & Pilipovic, M. (2019). Object detection and object tracking in front of the vehicle using front view camera. In *2019 Zooming Innovation in Consumer Technologies Conference (ZINC)* (pp. 27-32). IEEE Publishing. <https://doi.org/10.1109/ZINC.2019.8769367>
- Ding, X., & Yang, R. (2019). Vehicle and parking space detection based on improved YOLO network model. *Journal of Physics: Conference Series*, *1325*, Article 012084. <https://doi.org/10.1088/1742-6596/1325/1/012084>
- Dixit, K. S., Chadaga, M. G., Savalgimath, S. S., Rakshith, G. R., & Kumar, M. N. (2019). Evaluation and evolution of object detection techniques YOLO and R-CNN. *International Journal of Recent Technology and Engineering (IJRTE)*, *8*(2S3), 824-829. <https://doi.org/10.35940/ijrte.B1154070782S319>
- Gad, A. F. (2020). *Faster R-CNN explained for object detection tasks*. PaperspaceBlog. <https://blog.paperspace.com/faster-r-cnn-explained-object-detection/>.
- Giroto, E., Andrade, S., M. D., & Gonzalez, A. D. (2016). Professional experience and traffic accidents/ near-miss accidents among truck drivers. *ELSEVIER: Accidents Analysis and Prevention*, *95*(Pt A), 299-304. <https://doi.org/10.1016/j.aap.2016.07.004>
- Huang, Y. Q., Zheng, J. C., Sun, S. D., Yang, C. F., & Liu, J. (2020). Optimized YOLOv3 algorithm and its application in traffic flow detections. *Applied Sciences*, *10*(9), Article 3079. <https://doi.org/10.3390/app10093079>
- Johnson, K. D., Patel, S. R., Baur, D. M., Edens, E. Sherry, P. Malhotra, A., & Kales, S., N. (2014). Association of sleep habits with accidents and near misses in United States transportation operators.

- Journal of Occupational & Environmental Medicine*, 56(5), 510-515. <https://doi.org/10.1097/JOM.0000000000000132>
- Kataoka, H., Suzuki, T., Oikawa, S., Matsui, Y., & Satoh, Y. (2018). Drive video analysis for the detection of traffic near-miss incidents. In *IEEE International Conference on Robotics and Automation (ICRA)* (pp. 3421-3428). IEEE Publishing. <https://doi.org/10.1109/ICRA.2018.8460812>.
- Ke, R., Lutin, J., Spears, J., & Wang, Y. (2017). A cost-effective framework for automated vehicle-pedestrian near-miss detection through onboard monocular vision. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (pp. 25-32). IEEE Publishing.
- Mahdi, N. N. R. N., Bachok, N., Mohamed, N., & Shafei, M. N. (2014). Risk factors for near miss incident among long distance bus drivers in Malaysia. *Iranian Journal of Public Health*, 43(3), 117-124.
- Makizako, H., Shimada, H., Hotta, R., Doi, T., Tsutsumimoto, K., Nakakubo, S., & Makino, K. (2018). Associations of Near-Miss traffic incidents with attention and executive function among older Japanese drivers. *Gerontology*, 64, 495-502. <https://doi.org/10.1159/000486547>
- Matsui, Y., Hitosugi, M., Doi, T., Oikawa, S., Takahashi, K., & Ando, K. (2013). Features of pedestrian behavior in car-to-pedestrian contact situations in Near-Miss incidents in Japan. *Traffic Injury Prevention*, 14(1), 58-63. <https://doi.org/10.1080/15389588.2013.796372>
- Matsui, Y., Takahashi, K., Imaizumi, R., & Ando, K. (2011). Car-to-pedestrian contact situations in near-miss incidents and real-world accidents in Japan. In *22nd International Technical Conference on the Enhanced Safety of Vehicles* (No. 110164). National Traffic Safety and Environment Laboratory.
- Nadai, S. D., Parodi, F., & Pizzorni, D. (2012). A system of systems to Near Miss accidents in dangerous goods road transportation. In *IEEE 2012 7th International Conference on System of Systems Engineering (SoSE)* (pp. 219-222). IEEE Publishing. <https://doi.org/10.1109/SYSoSE.2012.6384198>.
- Nostikasari, D., & Shelton, K. (2017). *Learning from close calls: A glimpse into near-miss experiences*. Rice University Kinder Institute of Urban Research. <https://doi.org/10.25611/e51k-cpxo>
- Ong, Y. (2020). *Near miss vehicle collisions estimation using YOLO* (Master's dissertation). Universiti Sains Malaysia, Malaysia.
- Poulos, R., G., Hatfield, J., Rissel, C., Flack, L., K., Shaw, L., Grzebieta, R., & McIntosh, A., S. (2017). Near miss experiences of transport and recreational cyclists in New South Wales, Australia. Findings from a prospective cohort study. *Accidents Analysis and Prevention*, 101, 143-153. <https://doi.org/10.1016/j.aap.2017.01.020>
- Qin, Z., Yu, F., Liu, C., & Chen, X. (2018). How convolutional neural networks see the world - A survey of convolutional neural network visualization methods. *Mathematical Foundations Computing*, 1(2), 149-180. <https://doi.org/10.48550/arXiv.1804.11191>
- Rahman, A., Salam, A., Islam, M., & Sarker, P. (2012). An image based approach to compute object distance. *International Journal of Computational Intelligence Systems*, 1(4), 304-312. <https://doi.org/10.1080/18756891.2008.9727627>
- Rawat, W., & Wang, Z. (2017). Deep convolutional neural networks for image classification: A comprehensive review. *Neural Computation*, 29(9), 2352-2449. https://doi.org/10.1162/neco_a_00990

- Rome, L. D., Brown, J., Baldock, M., & Fitzharris, M. (2018). Near-miss crashes and other predictors of motorcycle crashes: Findings from a population-based survey. *Traffic Injury Prevention, 19*(2), S20-S26. <https://doi.org/10.1080/15389588.2018.1536822>
- Sanders, R., L. (2015). Perceived traffic risk for cyclists: The impact of near miss and collision experiences. *Accidents Analysis and Prevention, 75*, 26-34. <https://doi.org/10.1016/j.aap.2014.11.004>
- Silva Consultants. (2016). *Using near miss reporting in security*. Silva Consultants. <https://www.silvaconsultants.com/new-security-tips/using-near-miss-reporting-in-security>
- Siregar, M. L., Agah, H. R., & Hidayatullah, F. (2018). Near-miss accident analysis for traffic safety improvement at a “channelized” junction with U-turn. *International Journal of Safety and Security Engineering, 8*(1), 31-38. <https://doi.org/10.2495/SAFE-V8-N1-31-38>
- Sonnleitner, E., Barth, O., Palmanshofer, A., & Kurz, M. (2020). Traffic measurement and congestion detection based on real-time highway video data. *Applied Sciences, 10*(18), Article 6270. <https://doi.org/10.3390/app10186270>
- Srivastava, S., Divekar, A., V., Anilkumar, C., Naik, I., Kulkarni, V., & Pattabiraman, V. (2021). Comparative analysis of deep learning image detection algorithms. *Journal of Big Data volume, 8*(66), 1-31.
- Storgard, J., Erdogan, I., Lappalainen, J., & Tapanien, U. (2012). Developing incident and near miss reporting in the maritime industry - A case study on the Baltic Sea. *Procedia Social and Behavioral Sciences, 48*, 1010-1021. <https://doi.org/10.1016/j.sbspro.2012.06.1078>
- Uchida, N., Kawakoshi, M., Tagawa, T., & Mochida, T. (2010). An investigation of factors contributing to major crash types in Japan based on naturalistic driving data. *International Association of Traffic and Safety Sciences (IATSS), 34*(1), 22-30. <https://doi.org/10.1016/j.iatssr.2010.07.002>
- Vinitha, V., & Velantina, V. (2020). Social distancing detection system with artificial intelligence using computer vision and deep learning. *International Research Journal of Engineering and Technology (IRJET), 7*(8), 4049-4053.
- Wang, C., Dai, Y., Zhou, W., & Geng, Y. (2020). A vision-based video crash detection framework for mixed traffic flow environment considering low-visibility condition. *Hindawi, Journal of Advanced Transportation, 2020*, Article 9194028. <https://doi.org/10.1155/2020/9194028>
- WHO. (2015). *World Health Statistics 2015*. World Health Organization <https://www.who.int/docs/default-source/gho-documents/world-health-statistic-reports/world-health-statistics-2015.pdf>
- Zhang, Z., Trivedi, C., & Liu, X. (2018). Automated detection of grade-crossing-trespassing near misses based on computer vision analysis of surveillance video data. *Safety Science, 110*(Part B), 276-285. <https://doi.org/10.1016/j.ssci.2017.11.023>
- Zohra, A., F., Kamilia, S., & Souad, S. (2018). Detection and classification of vehicles using deep learning. *International Journal of Computer Science Trends and Technology (IJCST), 6*(3), 23-29.

Appendix A
 Supplementary Table
Previous research comparison of near miss

No	Researchers	Type of paper		Application data			Type of method				Type of accidents				Remarks
		Review	Research	Image process	Empirical data	Survey / Questionnaire	Model	Software	Others	Vehicles	Pedestrians	Motorcycles/ Bicycles			
1	Uchida et al., (2010)	✓	✓	✓		✓	✓		✓			✓			<ul style="list-style-type: none"> • Driving data acquisition system • Japan
2	Nadai et al., (2012)	✓	✓	✓		✓	✓								<ul style="list-style-type: none"> • System of Systems architecture (SoSE) • Biometric data • Can-Bus data • Italy
3	Storgard et al., (2012)	✓	✓	✓		✓	✓								<ul style="list-style-type: none"> • Logistics model • Wald Test • Baltic Sea
4	Matsui et al., (2013)	✓	✓	✓		✓	✓		✓	✓		✓			<ul style="list-style-type: none"> • Time to collision (TTC) • Japan
5	Matsui et al., (2011)	✓	✓	✓		✓	✓		✓	✓					<ul style="list-style-type: none"> • Pedestrian time-to-vehicle • vehicle time-to-collision • Japan
6	Johnson et al., (2014)	✓	✓	✓		✓	✓		✓						<ul style="list-style-type: none"> • Regression analyses • USA
7	Mahdi et al., (2014)	✓	✓	✓		✓	✓								<ul style="list-style-type: none"> • Multiple logistic regression • Malaysia
8	Aldred & Crossweller, (2015)	✓	✓	✓		✓	✓			✓		✓			<ul style="list-style-type: none"> • KeySurvey software • UK
9	Sanders, (2015)	✓	✓	✓		✓	✓					✓			<ul style="list-style-type: none"> • Internet survey • USA

Supplementary Table (continue)

No	Researchers	Type of paper		Application data		Type of method					Type of accidents				Remarks
		Review	Research	Image process	Empirical data	Survey / Questionnaire	Model	Software	Others	Vehicles	Pedestrians	Motorcycles/ Bicycles			
10	Aldred, (2016)	✓			✓	✓									<ul style="list-style-type: none"> • Semi-structured interview • UK
11	Giroto et al., (2016)		✓		✓	✓			✓						<ul style="list-style-type: none"> • Self-applied questionnaire • Brazil
12	Bull et al., (2017)		✓		✓		✓		✓						<ul style="list-style-type: none"> • Swedish Traffic Conflict Technique (TCT) • USA • TTC • USA
13	Calles et al., (2017)		✓		✓		✓		✓						<ul style="list-style-type: none"> • Histograms of Oriented Gradients (HOG) pedestrian detector • TTC • USA
14	Ke et al., (2017)		✓		✓		✓								<ul style="list-style-type: none"> • Questionnaire form • USA
15	Nostikasari & Shelton, (2017)		✓		✓		✓								<ul style="list-style-type: none"> • Questionnaire form • USA
16	Poulos et al., (2017)		✓		✓		✓								<ul style="list-style-type: none"> • Questionnaire form • Australia
17	Kataoka et al., (2018)		✓		✓			✓							<ul style="list-style-type: none"> • Near-Miss Incident Database (NIDB) • Japan
18	Makizako et al., (2018)		✓		✓		✓			✓					<ul style="list-style-type: none"> • Interview method • Japan
19	Rome et al., (2018)		✓		✓		✓								<ul style="list-style-type: none"> • Survey logistic regression • Australia

Supplementary Table (continue)

No	Researchers	Type of paper		Application data		Type of method			Type of accidents				• Remarks
		Review	Research	Image process	Empirical data	Survey / Questionnaire	Model	Software	Others	Vehicles	Pedestrians	Motorcycles/ Bicycles	
20	Siregar et al., (2018)	✓	✓	✓	✓			✓	✓	✓			<ul style="list-style-type: none"> • Fervor application • Indonesia
21	Zhang et al., (2018)	✓	✓	✓			✓		✓				<ul style="list-style-type: none"> • Region of interest (ROI) • Grade-crossing-trespassing near miss
22	Wang et al., (2020)	✓	✓	✓			✓	✓	✓	✓	✓		<ul style="list-style-type: none"> • USA • Retinex Image Enhancement Algorithms • YOLOv3 • China

