Vehicle Types Recognition in Night-Time Scene

Willy Liew[1](✉), Mohd Haris Lye Abdullah[1](✉), Rehan Shahid[2], and Amr Ahmed[2]

[1] Faculty of Engineering, Multimedia University, Persiaran Multimedia, 63100 Cyberjaya,
Selangor, Malaysia
1171101719@student.mmu.edu.my, haris.lye@mmu.edu.my

[2] Tapway, Pacific Place, Block C, Commercial Centre, C-3-1, Jalan PJU 1a/4, Ara Damansara,
47301 Petaling Jaya, Selangor, Malaysia

Abstract. Vehicle type recognition in night-time scene is a challenging issue to be resolved due to insufficient luminance, complex lighting environment in night-time and scarcity of public night-time vehicle dataset. Hence, in this paper, we analyse and evaluate the performance of several state-of-the-art model architectures including Faster R-CNN, YOLO and SSD for vehicle detection in night-time scene. Through comparison of evaluation metrics, YOLOv3 with DarkNet-53 achieves the best trade-off between detection accuracy and model architecture complexity, with Average Precision (AP) of 87.43%, recall rate of 91.48% and processing speed of 13.06 FPS with UA-DETRAC validation dataset. In addition, daytime to night-time image augmentation techniques through Neural Style Transfer (NST), conditional GAN (cGAN) and Cycle-Consistent Adversarial Networks (CycleGAN) are implemented to increase the number of night-time images for training dataset by translating the daytime images into night-time scene. Among the three approaches, CycleGAN can generate realistic and natural synthesized night-time images which contribute to improving detection accuracy of the vehicle type recognition model from mAP of 91.81% to 96.47%. Finally, we implement multiple objects tracking technique with Deep SORT algorithm to perform vehicle counting.

Keywords: Vehicle recognition · Object detection model · Generative adversarial network (GAN)

1 Introduction

Vehicle type recognition plays a vital role in Intelligent Transportation System (ITS) for smart city concept. ITS development has become the major driving force in enhancing economic growth, ensuring resilience of cities and improving urban-rural linkages. Some real-life applications of vehicle types of recognition include traffic flow analysis, toll fare collection, automated road speed enforcement and smart parking management.

Vehicle types recognition in nighttime scene is challenging especially when compared with daytime scene as the vehicle has lower luminance due to insufficient illumination and lack of contrast between vehicle and background in appearance information [1]. Furthermore, complicated lighting environment including interference from surrounding light sources such as streetlights, building illumination and reflection of lights from

M. Y. bin Alias (Ed.): MECON 2022, AER 214, pp. 139–153, 2023.
https://doi.org/10.2991/978-94-6463-082-4_15

vehicles are always confused with vehicles' head lamps and rear lamps by the detection model. Besides, public night-time vehicle dataset which is readily labelled and classified is scarce as compared to daytime vehicle dataset [2]. In order to achieve real time and high accuracy recognition, the proposed model has to trade-off between computational speed and complexity of model architecture.

There are three objectives in this project. Firstly, analysis, evaluation and comparison of different state-of-the-art object detection models, namely YOLO, Faster R-CNN and SSD with various backbone network including ResNet, MobileNet and DarkNet. The aim is to understand their respective architectures, performances, advantages and limitations. Secondly, implement advanced data augmentation techniques through day-time to night-time scene translation with generative adversarial models (GAN) models, namely Neural Style Transfer (NST), conditional GAN (cGAN) and CycleGAN. The synthesized images are used to increase the training images to cater for the scarce nigh-time images. Experiment is conducted to determine the quality of the generated images and impact of the generated images on detection accuracy. Lastly, vehicle tracking and counting system is deployed for post-processing and apply in real life scenario.

2 Literature Review

Existing computer vision and deep learning-based night-time vehicle type recognition methodologies can be generally classified into four main categories, which are motion-based detection, vehicle lamp recognition, deep neural network for vehicle detection and image enhancement techniques.

2.1 Motion-Based Detection

W. Zhang et al. [3] has proposed an unified approach to combine conventional three-frame difference with deep CNN (DCNN) to perform vehicle detection. The DCNN deployed in the proposed method is Overfeat [4], which reused a single CNN framework and shared feature learning base in performing localization, detection and classification at the same time. Q. Zou et al. [5] has implemented headlights detection, tracking and grouping to construct a robust nighttime vehicle detection system. It has combined context-based multiple object tracking with motion-based pairing of vehicle lamps to utilize the coherence between spatial and temporal components in improving detection accuracy.

Motion-based detection performs tracking of moving vehicles through computation of pixel differences of adjacent video frames. However, these approaches are not robust due to the uneven distribution of brightness, partial occlusion, reflections and complex lighting environment in the nighttime images which lead to unstable detection. This approach requires to solve multiple dimensional data association problem to perform object tracking which face difficulty due to noisy detection and ambiguity in data association. Finally, it has lower processing speed due to additional computation step required in vehicle detection system.

2.2 Vehicle Lamp Recognition or Car Face Segmentation

Z. Ding et al. [6] has proposed lamp pairs distance and contour information recognition (LDPC) to recognize vehicle types and sub-types effectively by using the composite features of lamp pairs and lamp contour as they are the intrinsic attributions in discriminating between various vehicle types. C. Chen et al. [7] has proposed multiple branches and multiple layer features vehicle detection method which focuses on capturing texture information from the car-face image and extracting the global and local features through Multi-branch CNN to recognize vehicle types.

The proposed approaches are relying on the intrinsic attributes and distinguishable characteristics of a vehicle to perform the vehicle type recognition. However, this approach is heavily relying on the detection result of the specific vehicle parts where their visibility and outline saliency are susceptible to environmental light interference, reflection, partial visibility of vehicle parts and occlusion from surrounding objects. Besides, computation through Multi-branch CNN will take N times longer processing time as compared to a single CNN model, where N is the number of branches integrated as each branch of CNN is an independent module.

2.3 Deep Neural Network

Q. Fan et al. [8] has conducted a comprehensive analysis on Faster R-CNN's underlying model structure to provide a more comprehensive understanding on strategy for tuning and modifying Faster R-CNN for specific objective and dataset and achieves remarkable improvement on the detection accuracy of Faster R-CNN compared to default setting. G. Xiaoying et al. [9] has proposed an advanced and modified SSD algorithm for vehicle detection to tackle the issue of missed detection and low accuracy. The proposed method implemented ResNet50 as the backbone network, integrated the semantic information from deep layers with position information from shallow layers and included Squeeze-and-excitation networks (SENet) in feature extraction layer. Y. Miao et al. [10] has implemented an robust night-time vehicle detection model which is based on YOLOv3 network. Multi Scale Retinex (MSR) is implemented to enhance the night-time image by improving the detailed feature representation. H.K. Leung et al. [2] has achieved effective detection performance with optimized Faster R-CNN model under extreme illumination conditions and compared the performance of VGG16 and ResNet101 for feature extraction.

Deep neural network can be generally classified as two-stage and single-stage methods. Faster R-CNN has decent and satisfactory overall detection accuracy, but it requires long processing time, SSD has the fastest processing speed as it has light model architecture, but it sacrifices the detection accuracy while YOLO is able to achieve trade-off between accuracy and speed, enabling it to achieve high mAP without sacrificing its FPS. These inferences are validated by the research work of Kim et al. [11].

2.4 Image Enhancement Techniques

C.T. Lin et. al [12] has proposed AugGAN, a GAN-based data augmenter and structure-aware unpaired image-to-image translation network which could perform transformation on images to desired domain. This method ensures that the image-objects attributes are well preserved and greatly reducing the artifacts in transformed images. X. Shao et al. [1] has proposed a cascaded detection model framework, which is named as FteGanOd. The model uses feature translation enhancement module, generative adversarial network and object detection algorithm. P. Tao et. al [13] has proposed novel BITPNet for night-time image enhancement which outperforms other competing low-light image enhancement approaches as evaluated by the proposed no-reference image quality metrics and visual quality.

Image enhancement method performs image scenes translation from daytime to nighttime scene or vice versa. It can either be used to increase the number of nighttime images dataset, as proposed in [12], or used to enhance the vehicles' features and improve detection accuracy as proposed in [1, 13]. However, this will lead to adverse effect of longer processing time due to additional process in the detection algorithm.

3 Methodology

The details of project implementation correspond to the three objectives defined earlier to ensure that the proposed approaches are able to solve the mentioned problem statements.

3.1 Dataset Preparation

Three different vehicle type recognition datasets have been described in the literature, namely BIT [14], UA-DETRAC [15] and CompCars [16]. The UA-DETRAC dataset is the most suitable for our project since the dataset provides different vehicle viewpoints. This includes front and rear face of vehicles. Besides, the environment is relatively more "noisy" and "chaotic" with the appearance of several background objects in various environment. Furthermore, the size of vehicle in the images is relatively small and it contains occluded vehicles. In addition to the public dataset, a custom private dataset known as Tapway vehicle dataset is used for evaluation of data augmentation using GAN technique.

The experiment conducted uses two datasets namely the UA-DETRAC dataset [15] and Tapway vehicle dataset. UA-DETRAC dataset comprises 100 videos which is divided into 60 sequences for training set and 40 sequences for testing set. The videos are recorded in both day and night-time, at 24 different places which depict different traffic conditions and patterns such as traffic crossings, T-junctions and urban highway, as shown in Fig. 1. There are four types of labelled vehicle, namely car, van, bus and others which include more vehicle types including tankers and trucks. The videos are recorded at 25 frames per second with jpeg image resolution 960x540.

The Tapway vehicle dataset is provided by Tapway company in this collaboration project. It shows various type of vehicle stopping at the toll booth in both day and night-time scene. The dataset is used to evaluate our proposed daytime to night-time scene translation model. Figure 2 shows that the dataset has 17 vehicle classes and can be categorised as daytime or night-time images. The dataset has significant imbalance between number of daytime and nighttime images with ratio of 6.4 to 1. Besides, it also has significant imbalance between number of images for different vehicle classes.

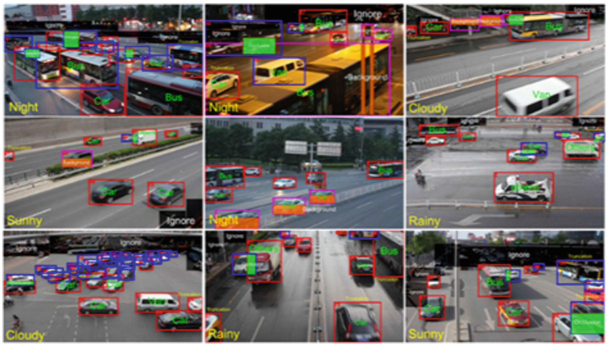


Fig. 1. Example of UA-DETRAC dataset

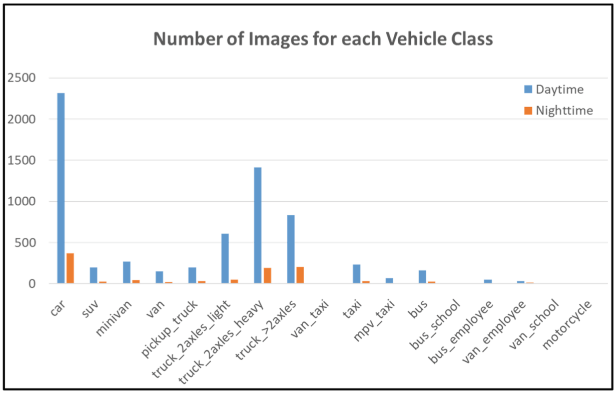


Fig. 2. Original distribution of Tapway dataset

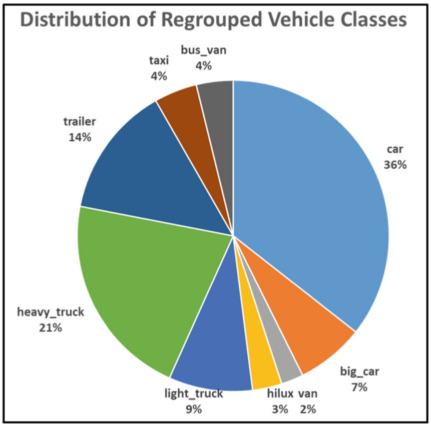


Fig. 3. Distribution of regrouped vehicle classes

Regrouping of vehicle classes is done to reduce the number of classes from 17 to 9 by merging according to the vehicle classes in toll fare collection. The distribution of vehicle classes is more balanced after regrouping, as shown in Fig. 3. However, in order to solve the imbalance issues between day and night-time samples in the dataset, advanced data augmentation technique is proposed and implemented.

3.2 Object Detection Model

Object detection model predicts bounding box and its class label. It uses backbone network for feature extraction. The feature is then used to predict to location of the object in the image. Object detection algorithms can be generally categorised as one-stage or two stage method. Single stage method makes a fixed number of predictions with predefined grid cells. Example of single stage detector includes YOLO [17] and SSD [18].

Two-stage method implements a proposal network to find candidate image regions that can possibly contain the target object. The candidate region is then classified to determine the object type. Such method is implemented in Faster R-CNN [19]. Transfer learning uses pre-trained model to enable the model to converge faster and reduce number of training step required. Four different object detection models have been built with pre-trained weights that are trained with COCO 2017 dataset as shown below:

- SSD MobileNet V2 FPNLite 320x320 [20]
- SSD ResNet50 V1 FPN 640x640 (RetinaNet50) [20]
- Faster R-CNN ResNet50 V1 640x640 [20]
- YOLOv3 DarkNet-53 416x416 [21].

In the training process, Stochastic Gradient Descent (SGD) with momentum optimizer is implemented to accelerate training process by including past updates in backward propagation to dampen the changes of gradient, reduce oscillation and avoid being stuck at the local minima. Besides, L2 regularizer is implemented to ensure the model is able to generalize to new domain of dataset and prevent overfitting. Moreover, learning rate scheduler is implemented to control the magnitude of change for each gradient descent. Through experimental results, we found out that cosine decay with warmup worked best for ResNet50 model while cosine decay without warmup is more suitable for MobileNetV2 and DarkNet-53 models, as shown in Fig. 4.

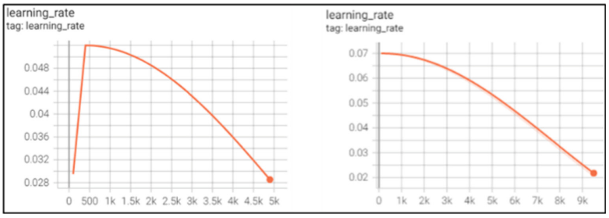


Fig. 4. Cosine decay learning rate with and without warmup

3.3 Data Augmentation Techniques

Data augmentation is an important strategy to overcome insufficient data diversity and data scarcity challenges. The method generates artificial versions of the real image to increase dataset size. However, as the major problem is the imbalance number of night-time image compared to daytime image, common data augmentation techniques such as cropping, rotation and scaling are not useful. Hence, three advanced data augmentation techniques which are capable of translating daytime images to nighttime scene are deployed to enlarge training dataset and improve the detection accuracy.

Neural style transfer (NST) [22] is an optimization technique which composes generated image to simultaneously match the style statistics of style reference image with the content statistics of content image through matching of the Gram matrix statistics from pre-trained deep features. However, the major issue is NST has slow optimization process which will take significant long time to train the style transfer model for our use case Hence, pre-trained arbitrary image stylization model, which is constructed with reference to [29], is implemented instead as the style transfer network.

Conditional adversarial network (cGAN) [23] learns a mapping from input images to output images for general-purpose solution to image-to-image translation problems. The major advantage of cGAN as compared to NST is it does not require to hand engineer the mapping functions and loss functions corresponding to different use case and scenario. It learns structured loss where any possible different in structure between output and target is being penalized. The architecture comprises one U-Net-architecture-based generator and one convolutional PatchGAN classifier-based discriminator.

Cycle-Consistent adversarial networks (CycleGAN) [24] captures the unique characteristics and attributes of one image domain and then translates them into other image domain, without requiring paired training images as dataset. It can perform unpaired image-to-image translation. The training of CycleGAN uses total generator loss with two additional loss functions, namely cycle consistency loss and identity loss. Figure 5 shows the model architecture of CycleGAN.

3.4 Vehicle Tracking and Counting System

Tracking and association are two important elements in object tracking in video. Tracking involves the process of estimating the location and position of the same object over continuous video sequences while association of frame sequences is the matching of detections in previous frame with current frame through ID assignment. Deep SORT

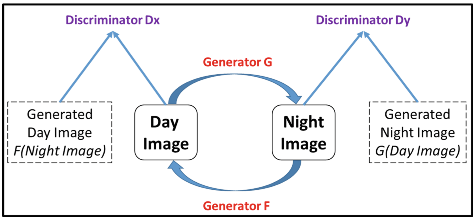


Fig. 5. Model architecture of CycleGAN

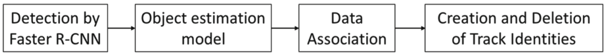

Detection by Faster R-CNN
Object estimation model
Data Association
Creation and Deletion of Track Identities

Fig. 6. Key components of SORT

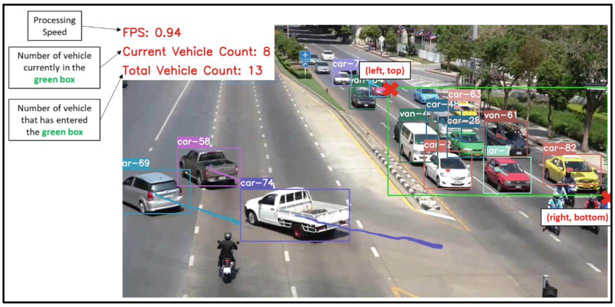


Fig. 7. Example of output from vehicle counting system

[25] (Simple Online and Realtime Tracking) as shown in Fig. 6 is implemented for vehicle tracking and counting system as it has better performance in tracking occluded objects and reducing number of identity switches as compared to SORT [26].

The vehicle counting process is visualized in Fig. 7. The green box is a region of interest that is used to determine the density of vehicles. It can be customized by specifying the left, top, right, bottom coordinates. The green box can be transformed into a single line by simply changing the top coordinate to be equal to the bottom coordinate. There are three important results generated, namely the processing speed in term of FPS, number of vehicles that are currently inside the green box and number of vehicles that has passed the green box.

4 Results and Discussions

Experimental results and critical analyses are carried out to evaluate and validate the methodologies proposed.

4.1 Performance of Object Detection Models

The object detection models are trained with UA-DETRAC dataset [15]. The model performance is evaluated based on Precision-Recall curve, Average Precision (AP) and processing speed in frames per second (FPS). Precision-recall (PR) curves and Average Precision (AP) are generated with the algorithm proposed by R. Padilla [27]. Figure 8 shows that YOLOv3 model has the largest Area under Curve (AUC) and highest AP which is 87.43% while SSD MobileNetV2 had the poorest performance due to its lowest AP which is 76.73%.

Table 1 shows that SSD MobileNetV2 has the highest frame per second (FPS) which implies that it has the fastest processing speed. The YOLOv3 model has the highest recall rate which means it can retrieve most of relevant objects available in the dataset.

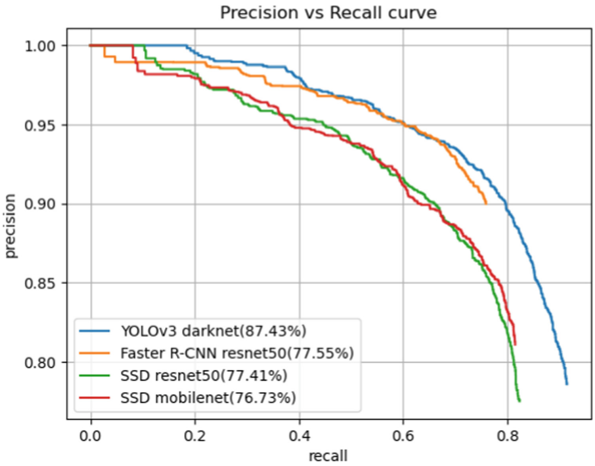


Fig. 8. Precision recall curve of trained object detection models on UA-DETRAC dataset

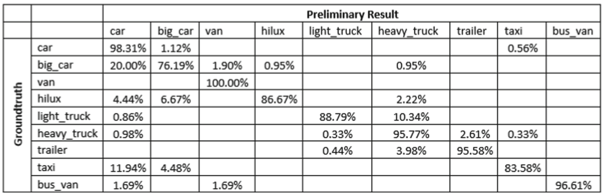
Table 1. Evaluation metrics for trained object detection models

Detection Model	SSD		Faster R-CNN	YOLOv3
Backbone Network	Mobilenet V2	ResNet50	ResNet50	Darknet-53
Input size	320 x 320	640 x 640	640 x 640	416 x 416
Number of layers	53	50	50	53
Number of parameters	3.4 million	23.9 million	23.9 million	65.2 million
Processing speed	14.72 FPS	5.35 FPS	4.16 FPS	13.06 FPS
AP	76.73%	77.41%	77.55%	87.43%
Recall rate	81.53%	82.38%	75.92%	91.48%

Through the comparison of various performance metrics, we conclude that that YOLOv3 with DarkNet-53 is the best recognition model among the four trained model as it achieves the best trade-off between detection accuracy and processing speed. This is due to YOLOv3 one-stage method which detect object through a unified model architecture. The DarkNet-53 backbone provide good discriminative features for accurate object detection.

4.2 Performance of Data Augmentation Techniques

In order to validate the contribution of data augmentation techniques in increasing detection accuracy, preliminary result in terms of confusion matrix is generated by training and testing the YOLOv3 model with the Tapway dataset of 3761 training images for 13 epochs. The model is then tested with 1493 validation images. Figure 9 shows model performance as represented by the confusion matrix on the multi-class vehicle recognition task where augmentation is not being used. The performance based on recall rate is



		Preliminary Result								
		car	big_car	van	hilux	light_truck	heavy_truck	trailer	taxi	bus_van
Groundtruth	car	98.31%	1.12%						0.56%	
	big_car	20.00%	76.19%	1.90%	0.95%		0.95%			
	van			100.00%						
	hilux	4.44%	6.67%		86.67%		2.22%			
	light_truck	0.86%				88.79%	10.34%			
	heavy_truck	0.98%				0.33%	95.77%	2.61%	0.33%	
	trailer					0.44%	3.98%	95.58%		
	taxi	11.94%	4.48%						83.58%	
	bus_van	1.69%		1.69%						96.61%

Fig. 9. Confusion matrix (no augmentation)

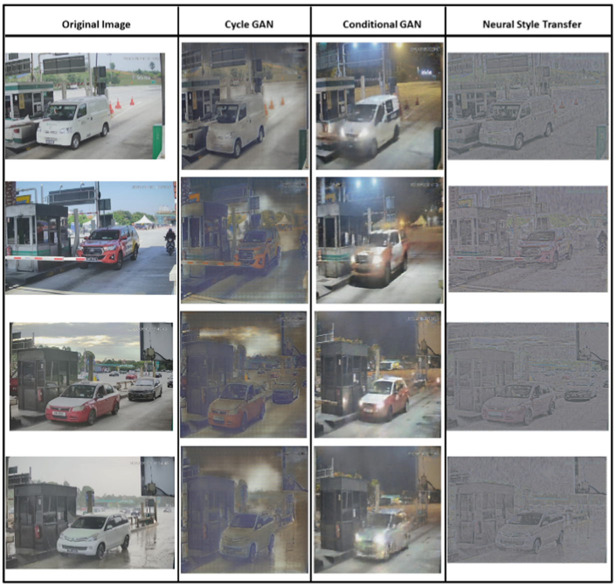
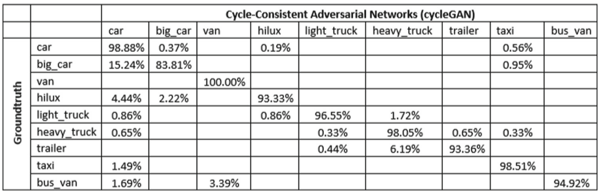


Fig. 10. Comparison of generated nighttime images

observed to be lower on three classes, namely “big_car” (76.19%), “taxi” (83.58%) and “hilux” (86.67%). This is due to the similarity of the classes with other related classes and small training samples.

In order to increase the number of night-time images, Generative Adversarial Model (GAN) models are used for day to night-time style transfer. Figure 10 shows the synthesized night-time images which were generated by the three approaches, namely CycleGAN, cGAN and NST. The result of NST generated image was not natural as it did not resemble night-time scene due to the model original use for artistic painting style transfer. The model by cGAN can preserve the vehicle texture and outline of daytime image without significant distortion. The model can produce synthesized night-time images which are more natural and resemble real world night-time environment. However, the cGAN is optimized to reconstruct and retain the attribute of night-time images that are used for training. As comparison, CycleGAN model can achieve optimized trade-off between night-time scene translation performance and ability to preserve vehicle texture and outline. This produces natural and realistic synthesized night-time images.



		Cycle-Consistent Adversarial Networks (cycleGAN)								
		car	big_car	van	hilux	light_truck	heavy_truck	trailer	taxi	bus_van
Groundtruth	car	98.88%	0.37%		0.19%				0.56%	
	big_car	15.24%	83.81%						0.95%	
	van			100.00%						
	hilux	4.44%	2.22%		93.33%					
	light_truck	0.86%			0.86%	96.55%	1.72%			
	heavy_truck	0.65%				0.33%	98.05%	0.65%	0.33%	
	trailer					0.44%	6.19%	93.36%		
	taxi	1.49%							98.51%	
	bus_van	1.69%		3.39%						94.92%

Fig. 11. Confusion matrix for training with CycleGAN generated images

Table 2. Summary of results with different techniques

	No augmentation	NST	cGAN	Cycle-GAN
mAP	91.81%	91.38%	92.44%	**96.47%**
Class accuracy	93.97%	93.97%	94.84%	**96.18%**
Validation loss	1.83	1.84	1.65	1.56
Average Precision (AP) for each vehicle class				
big_car	76.08%	67.77%	77.26%	**88.97%**
bus_van	**100%**	97.75%	99.92%	98.31%
car	**96.51%**	96.09%	96.35%	98.35%
heavy_truck	94.02%	95.85%	**96.68%**	96.00%
hilux	90.48%	92.82%	87.37%	**99.86%**
light_truck	90.72%	**96.62%**	96.62%	96.43%
taxi	84.37%	85.85%	91.27%	**97.08%**
trailer	95.00%	92.88%	**95.80%**	94.37%
van	**99.11%**	96.80%	90.72%	98.86%

The result on using synthetic images generated from CycleGAN model is shown in Fig. 11. The confusion matrix result shows improved accuracy for selected classes. The low performance class as observed in Fig. 9 namely “big car”, “taxi” and “hilux” have shown improved accuracy as shown in Fig. 11. This shows that training with CycleGAN generated image has contributed significantly in improving the detection accuracy of the model.

Table 2 shows the comparison of detection accuracy among the three image generation methods. CycleGAN generated image contributed the highest mean average precision (mAP) and class prediction accuracy. The average precision for individual class shows satisfactory value. The underlined value shows the best result for the performance metric used.

4.3 Performance of Vehicle Counting System

In order to determine the accuracy of the vehicle counting system, we have recorded a few night-time highway videos at Malaysia’s highway on the pedestrian bridges as

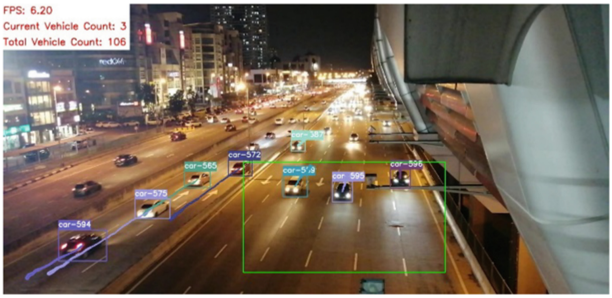


Fig. 12. Output of vehicle counting system at IOI Puchong

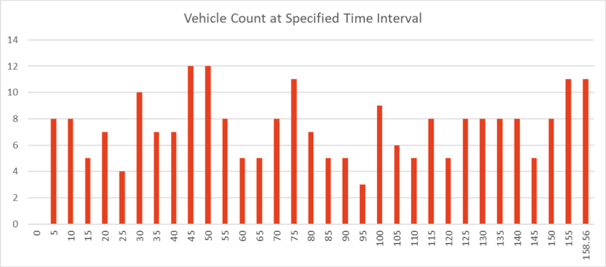


Fig. 13. Vehicle count at IOI Puchong

Table 3. Validation of vehicle counting result

Video	Length of Video	System Result	Manual Counting	Difference	Error
Lotus Puchong	127.16 s	124	120	+ 4	3.33%
IOI Puchong	158.56 s	229	220	+ 9	3.93%

test videos. In Fig. 12**,** the night scene comprised of complex lighting environment with interference from streetlight, building illumination and vehicle head lamps. This imposes challenges for accurate vehicle detection.

Figure 13 shows the record of vehicle count which is updated in an interval of 5 s and can be customized depending on different scenario. This function can be used to determine the peak hour of the highway with congested traffic with high vehicle volume.

To validate the accuracy of vehicle counting system, manual counting is performed on the test videos to determine their difference in the count value. As shown in Table 3, the percentage of error for both videos are less than 4%. The vehicle counting system produced more vehicles count than the actual number as it occasionally loses track of the vehicle and cause the assignment of new track identity. This means the system treat the same vehicle as different entities due to different track identity.

5 Conclusions and Future Works

This paper proposes and evaluates vehicle type recognition in night-time scene. After comparing different state-of-the-art object detection models on the night time vehicle detection dataset, YOLOv3 with DarkNet-53 achieves the best trade-off between detection accuracy and execution speed. Through the evaluation of three different data augmentation techniques on the Tapway vehicle dataset, CycleGAN model achieves the best daytime to night-time scene translation performance as the generated images resemble the night-time scene and preserves vehicle appearance. Furthermore, generated images with CycleGAN provides good image augmentation and this contributes towards improved vehicle detection accuracy. Finally, vehicle tracking and counting system is implemented effectively with the Deep SORT tracking algorithm in the night-time scene.

Acknowledgments. We are grateful to the engineers from Tapway, which has provided the Malaysian toll booth vehicle detection dataset for in this collaboration project.

References

1. X. Shao, C. Wei, Y. Shen, and Z. Wang, “Feature Enhancement Based on CycleGAN for Nighttime Vehicle Detection,” *IEEE Access*, vol. 9, pp. 849–859, 2021, doi: https://doi.org/10.1109/ACCESS.2020.3046498.
2. H. K. Leung, X.-Z. Chen, C.-W. Yu, H.-Y. Liang, J.-Y. Wu, and Y.-L. Chen, “A Deep-Learning-Based Vehicle Detection Approach for Insufficient and Nighttime Illumination Conditions,” *Appl. Sci.*, vol. 9, no. 22, 2019, doi: https://doi.org/10.3390/app9224769.
3. W. Zhang, L. Chen, W. Gong, Z. Li, Q. Lu, and S. Yang, “An integrated approach for vehicle detection and type recognition,” in *Proceedings - 2015 IEEE 12th International Conference on Ubiquitous Intelligence and Computing, 2015 IEEE 12th International Conference on Advanced and Trusted Computing, 2015 IEEE 15th International Conference on Scalable Computing and Communications, 20*, 2016, pp. 798–801, doi: https://doi.org/10.1109/UIC-ATC-ScalCom-CBDCom-IoP.2015.157.
4. P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. Lecun, “OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks,” *Int. Conf. Learn. Represent.*, 2013.
5. Q. Zou, H. Ling, S. Luo, Y. Huang, and M. Tian, “Robust Nighttime Vehicle Detection by Tracking and Grouping Headlights,” *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2838–2849, 2015, doi: https://doi.org/10.1109/TITS.2015.2425229.
6. Z. Ding and W. Mo, “Vehicle Type and Sub-type Recognition by Lamp Pairs Distance and Lamp Contour in ITS,” in *2020 IEEE 8th International Conference on Computer Science and Network Technology (ICCSNT)*, 2020, pp. 134–138, doi: https://doi.org/10.1109/ICCSNT50940.2020.9304981.
7. C. Chen, X. Cai, Q. Zhao, L. Lv, and H. Shu, “Vehicle type recognition based on multi-branch and multi-layer features,” in *2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, 2017, pp. 2038–2041, doi: https://doi.org/10.1109/IAEAC.2017.8054374.
8. Q. Fan, L. Brown, and J. Smith, “A closer look at Faster R-CNN for vehicle detection,” in *2016 IEEE Intelligent Vehicles Symposium (IV)*, 2016, pp. 124–129, doi: https://doi.org/10.1109/IVS.2016.7535375.

9. G. Xiaoying, L. Qiaoling, Q. Zhikang, and X. Yan, “Target Detection of Forward Vehicle Based on Improved SSD,” in *2021 IEEE 6th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA)*, 2021, pp. 466–468, doi: https://doi.org/10.1109/ICCBDA51879.2021.9442550.
10. Y. Miao, F. Liu, T. Hou, L. Liu, and Y. Liu, “A Nighttime Vehicle Detection Method Based on YOLO v3,” in *2020 Chinese Automation Congress (CAC)*, 2020, pp. 6617–6621, doi: https://doi.org/10.1109/CAC51589.2020.9326819.
11. J. Kim, J.-Y. Sung, and S. Park, “Comparison of Faster-RCNN, YOLO, and SSD for Real-Time Vehicle Type Recognition,” in *2020 IEEE International Conference on Consumer Electronics - Asia (ICCE-Asia)*, 2020, pp. 1–4, doi: https://doi.org/10.1109/ICCE-Asia49877.2020.9277040.
12. C.-T. Lin, S.-W. Huang, Y.-Y. Wu, and S.-H. Lai, “GAN-Based Day-to-Night Image Style Transfer for Nighttime Vehicle Detection,” *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 2, pp. 951–963, 2021, doi: https://doi.org/10.1109/TITS.2019.2961679.
13. P. Tao, H. Kuang, Y. Duan, L. Zhong, and W. Qiu, “BITPNet: Unsupervised Bio-Inspired Two-Path Network for Nighttime Traffic Image Enhancement,” *IEEE Access*, vol. 8, pp. 164737–164746, 2020, doi: https://doi.org/10.1109/ACCESS.2020.3022393.
14. Z. Dong, Y. Wu, M. Pei, and Y. Jia, “Vehicle Type Classification Using a Semisupervised Convolutional Neural Network,” *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 2247–2256, 2015, doi: https://doi.org/10.1109/TITS.2015.2402438.
15. L. Wen *et al.*, “UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking,” *Comput. Vis. Image Underst.*, vol. 193, 2020, doi: https://doi.org/10.1016/j.cviu.2020.102907.
16. L. Yang, P. Luo, C. C. Loy, and X. Tang, “A large-scale car dataset for fine-grained categorization and verification,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 3973–3981, doi: https://doi.org/10.1109/CVPR.2015.7299023.
17. J. Redmon and A. Farhadi, “YOLOv3: An Incremental Improvement,” 2018.
18. Liu, W., et al.: SSD: single shot MultiBox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
19. K. He, X. Zhang, S. Ren, and J. Sun, “Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition,” *CoRR*, vol. abs/1406.4, 2014, [Online]. Available: http://arxiv.org/abs/1406.4729.
20. H. Yu *et al.*, “TensorFlow Model Garden.” 2020, [Online]. Available: https://github.com/tensorflow/models.
21. A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal Speed and Accuracy of Object Detection.” 2020, [Online]. Available: https://github.com/pjreddie/darknet.
22. L. A. Gatys, A. S. Ecker, and M. Bethge, “A Neural Algorithm of Artistic Style,” *CoRR*, vol. abs/1508.0, 2015, [Online]. Available: http://arxiv.org/abs/1508.06576.
23. P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-Image Translation with Conditional Adversarial Networks,” *CoRR*, vol. abs/1611.0, 2016, [Online]. Available: http://arxiv.org/abs/1611.07004.
24. J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks,” *CoRR*, vol. abs/1703.1, 2017, [Online]. Available: http://arxiv.org/abs/1703.10593.
25. N. Wojke, A. Bewley, and D. Paulus, “Simple Online and Realtime Tracking with a Deep Association Metric,” *CoRR*, vol. abs/1703.0, 2017, [Online]. Available: http://arxiv.org/abs/1703.07402.

26. A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple Online and Realtime Tracking," *CoRR*, vol. abs/1602.0, 2016, [Online]. Available: http://arxiv.org/abs/1602.00763.
27. R. Padilla, W. L. Passos, T. L. B. Dias, S. L. Netto, and E. A. B. Da Silva, "A comparative analysis of object detection metrics with a companion open-source toolkit," *Electron.*, vol. 10, no. 3, pp. 1–28, 2021, doi: https://doi.org/10.3390/electronics10030279.