An Improved YOLO Model for Vehicle Recognition System in Aerial Imagery

Md Abdul Momin[1], Mohamad Haniff Junos[2], Anis Salwa Mohd Khairuddin[1*], Mohamad Sofian Abu Talip[1], Akira Taguchi[3]

[1]Department of Electrical Engineering, Faculty of Engineering, Universiti Malaya, 50603, Kuala Lumpur, Malaysia

[2]School of Aerospace Engineering, Universiti Sains Malaysia, Engineering Campus, 14300, Nibong Tebal, Penang, Malaysia

[3]Department of Computer Science, Faculty of Information Technology, Tokyo City Universiti, Tokyo, 158-8557 Japan

[*]Corresponding Author
doi: https://doi.org/10.21467/proceedings.141.3

ABSTRACT

The modern development in unmanned aerial vehicles (UAV) providing aerial imagery attracts researchers to improve the object detection algorithms to be used in various applications. Lightweight object detection models are required for low computational resource devices. This study developed a lightweight object detection model by improving the architecture of YOLOv4 Tiny to detect vehicles from the VEDAI dataset. In the developed model, one additional scale feature map is added to the architecture. Besides that, the sizes of output images for the second and third prediction boxes are upscaled with the aim of detecting the small pixels of vehicles in the aerial imagery with better accuracy. The experimental results showed an improvement in the detection accuracy and precision when compared with several state-of-the-art methods to detect small objects such as vehicles in aerial imagery.

Keywords: Aerial, image processing, object detection, YOLO

1 Introduction

The advancement of computer vision has been an essential part of autonomous vehicles to detect vehicles in smart cities. In autonomous vehicles, recognising the vehicles around them is important for safe driving and tracking [1]. Computer vision applications on autonomous transport systems include vehicle counting, plate number recognition, vehicle flow prediction, traffic scene and vehicle speed measurement [2]. However, real-time vehicle detection in traffic scenes is challenging due to the complex background, occlusion error, and bad weather [3]. Wireless data transmission systems are used in autonomous vehicles to detect objects which delays the object detection process. Video and image processing requires high computational computers to detect objects [4]. However, the GPU of autonomous vehicles is not computationally efficient in detecting objects with good accuracy and precision, which makes researchers interested in vehicle recognition systems for autonomous vehicles.

With the development of drones and satellite technology, aerial imagery adds an important dimension to the field of autonomous transport systems to detect images. In aerial images, the objects are small since the objects are captured from different altitudes. Aerial photography using UAVs embellishes the remote sensing technology affordable and more available for implementation of projects in different research areas due to its uncomplicated manoeuvrability with good image resolution and suitable customisation of the algorithm for object detection [5]. A massive dataset from UAV imagery can be very tiresome for humans

Proceedings DOI: 10.21467/proceedings.141; Series: AIJR Proceedings; ISSN: 2582-3922; ISBN: 978-81-957605-4-1

to retrieve, view and process in object detection methods. Deep learning-based object detection algorithms provide consequential accuracy and precision for detecting objects for remote sensing images. Researchers are developing Convolutional Neural Networks (CNN), which are befitting for real-time object detection in autonomous vehicles, drones, or robotics, and they can detect the target objects with accuracy and precision. Computer vision-based object detection methods can be categorised into one-stage and two-stage methods [6]. Two-stage object detection methods have higher accuracy than one-stage object detection algorithms. However, one-stage object detection methods are more lightweight and can obtain real-time detection quickly [7]. In this study, a one-stage object detection method, You Only Look Once (YOLO) [8], will be improved to recognise vehicles for lightweight applications. This study aims to detect vehicles with better accuracy in aerial imagery where the vehicles are small, and the background is complex based on the improved YOLOv4 Tiny model.

2 Literature Review

Deep Convolutional Neural Network (DCNN) can extricate the deeper features of the remote sensing images automatically, giving advantages over superficial machine learning. The robust adaptability of CNN in object detection algorithms makes it the prime choice for aerial imagery classifications [9]. A single Convolutional Neural Network (CNN) network is used in a one-stage object detection method to predict the position and classification of the target objects. Single Shot Multibox Detector (SSD) [10], Learning Rich Feature (LRF) [11], RetinaNet, RefineDet, and You Only Look Once (YOLO) [8] series are the one-stage object detection methods where the algorithms are considered as regression problem explicitly. The lightweight version of these models is YOLOv1 tiny, YOLOv2 Tiny, YOLOv3 Tiny and YOLOv4 Tiny. YOLOv4 Tiny has a lower requirement of computing ability in the hardware. In YOLOv4 Tiny, the model parameters of YOLOv4 are reduced, and it decreases the architecture's computing complexity while ensuring moderate accuracy. The vehicle Detection in Aerial Imagery (VEDAI) dataset [12] has images of vehicles captured in uncontrolled environments with different orientations and abrupt changes in the background and the change of lighting and shadow with occlusion issues. Since the researchers are trying to improve the vehicle detection methods, some researchers used the VEDAI dataset to detect small vehicles from aerial imagery where the CNN was modified with the input size of the images was 512*512 and the percentage of mean average precision (mAP) was 47.8 [13]. Some authors run the Single Shot Multibox (SSD) algorithm on the VEDAI dataset with the image input size of 512*512 and found the mAP as 43.1%. The dataset provided a mAP of 46.9% when using the YOLOv2 object detection algorithm [14]. The onboard GPU in autonomous vehicles is not computationally efficient as high computational computers. The detection speed must be robust for detecting vehicles and traffic scenes in real-time to be used in autonomous transportation systems. Therefore, lightweight object detection models are indispensable for real-time object detection applications to provide the detected information with the least latency and error.

3 Research Methodology

This study aims to improve a YOLO algorithm that can detect vehicles from aerial images. The challenge of detecting objects from aerial imagery is the objects are very small with complex backgrounds as the images are captured from different altitudes. This section will discuss the dataset and the developed algorithms Model A and Model B based on YOLOv4 Tiny. In this study, the improved YOLO algorithm was tested on the VEDAI dataset. The total number of images was 1250 in this experiment, where 1125 were used for training, and 125 images were used for testing. Nine classes of objects were annotated, and the classes were 'car', 'truck', 'van', 'tractor', 'pickup', 'camping', 'plane', 'boat' and 'other'. The software

Proceedings DOI: 10.21467/proceedings.141
ISBN: 978-81-957605-4-1

Series: AIJR Proceedings
ISSN: 2582-3922

environment for training and testing were CUDA 11.1, CuDNN 7.6.5, and Open CV 4.1.0 for NVIDIA Geforce 940MX-based computers. In Google Collab, the Open CV version was 3.1.0, and other software was similar to GPU-based computers. The deep learning framework was darknet for this study. The input image size was 416x416 in the configuration files in this experiment.

The conventional YOLOv4 Tiny model was improved in this study to recognise vehicles. YOLOv4 Tiny is a one-stage regression-based object detection method. In the backbone of YOLOv4 Tiny, CSPDarknet-53 Tiny is adopted, and Cross Stage Partial (CSP) network improves the accuracy of the algorithm since it works as a bridge between the previous layer to the following layer to pass the image information [15]. There are three CSP blocks in the backbone of the YOLOv4 Tiny model, and each CSP block consists of four convolutional layers. LeakyReLu is used as an activation function in the convolutional layers of YOLOv4 Tiny, reducing the computational parameters [16]. It contains two prediction boxes to show the detected objects as output. The default image input size is 416x416, and the output is 13x13 and 26x26 in two feature maps to predict the detected objects [17]. Since the output size is very small, YOLOv4 Tiny does not provide efficient performance in detecting small objects on aerial imagery and results in occlusion errors for overlapping objects [15]. Therefore, this study improved the YOLOv4 Tiny model, and Model A and Model B were developed to test on the VEDAI dataset.

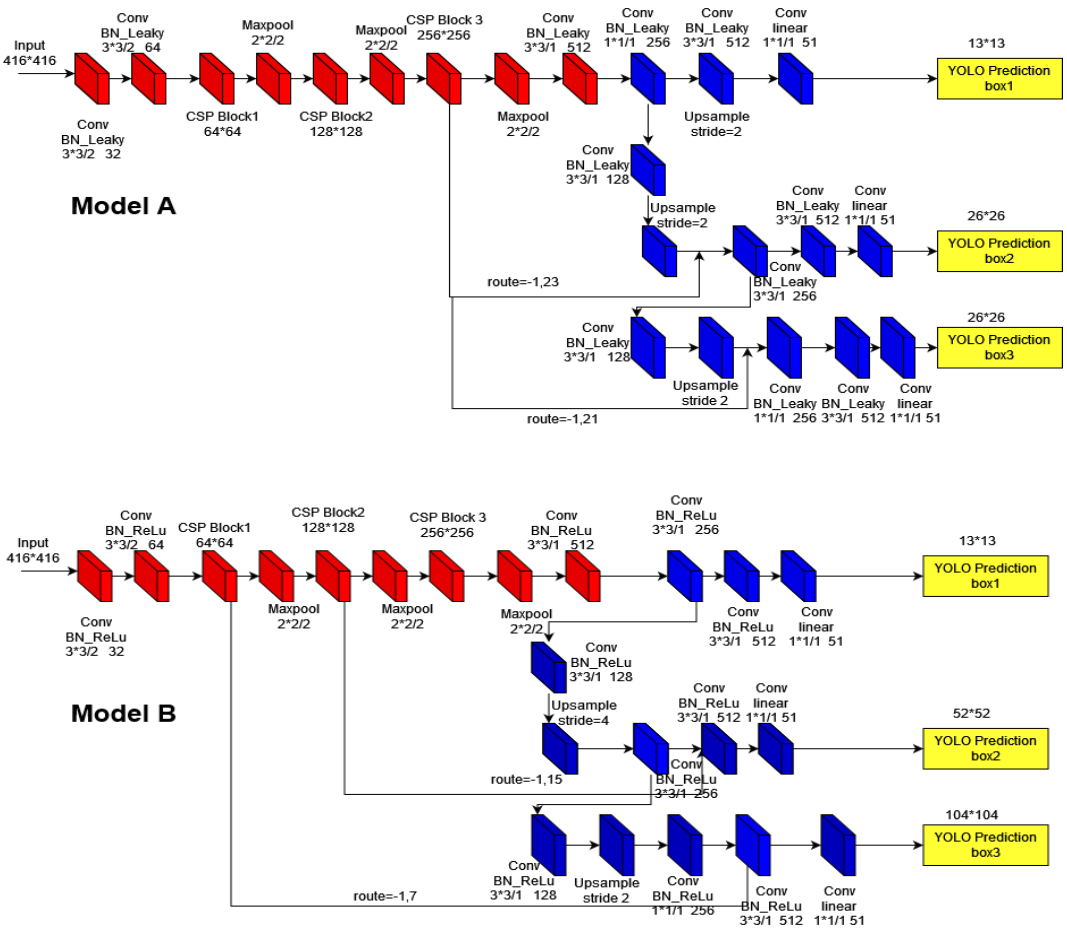


Figure 1: *Architectures of developed YOLO models for vehicle recognition*

The architectures of the developed models are shown in Figure 1. Model A and Model B are similar to the backbone of YOLOv4 Tiny. The main difference between YOLOv4 Tiny and the developed models is the number of prediction boxes. Both Model A and Model B have three prediction boxes. One additional

prediction box increases the computational complexity of the algorithm. However, the additional feature map scale increases the average precision in Model B. The output image sizes of Model A are 13x13, 26x26 and 26x26 in the three prediction boxes consecutively. In Model B, the output sizes of the architecture are 13x13, 52x52 and 104x104. The output sizes were increased to detect small objects in the aerial dataset. In Model A, the second and third prediction boxes were concatenated with the third CSP block of the backbone. The second and third prediction boxes were concatenated with the backbone's second and first CSP block sequentially in Model B. To increase the output size of the second scale feature map connected with the first prediction box, the upsample stride was doubled in Model B. The total number of layers in YOLOv4 Tiny is 38, 48 layers in Model A and 47 layers in Model B. After the training process, the weight file of Model A was 37.1 Mb and 30.9 Mb for Model B.

4 Results and Discussion

This section discusses the test results of developed models and different versions of YOLO models on the VEDAI dataset. This section focuses on comparing the test results, mean average precision and other object detection parameters. Table 1 illustrates the average precision of different classes of the VEDAI dataset on different YOLO and developed models. The conventional YOLOv3 and YOLOv4 provided better mean average precision than YOLOv4 Tiny and the developed models. However, to apply the models in the vehicle recognition systems in the autonomous transport systems, lightweight CNN models, which were Model A and Model B, were developed based on YOLOv4 Tiny. The highest mAP percentage was 73.38 by the YOLOv4 algorithm. Among the lightweight models, the highest percentage of mean average precision (mAP) was 50.8 on Model B, and the least was provided by Model A, which was 38.93. Model B showed a better percentage of mAP than YOLOv4 Tiny since the output image size of the second and third prediction boxes were 52x52 and 104x104, as a larger output image size improves the precision of detecting small vehicles in the aerial dataset. Model A showed a lower percentage of mAP than YOLOv4 Tiny and Model B, although there were three prediction boxes in Model A. The mAP percentage was lower due to the similar image output size of Model A's second and third prediction boxes. The third prediction box increased Model A's computational complexity and weight file but decreased the mAP percentage.

Table 1: *Average precision of YOLO and developed models on the VEDAI dataset*

Model	Car %	Truck %	Van %	Tractor %	Pickup %	Camping %	Plane %	Boat %	Other %	mAP %
YOLOv3	54.68	63.49	54.78	50	75.79	67.57	87.5	45.14	29.32	61.79
YOLOv4	63.37	89.04	65.62	82.75	51.49	86.87	100	76.36	35.67	73.38
YOLOv4 tiny	69.3	56.74	9.87	53.17	43.82	50.46	91.1	19.91	35.52	47.8
Model A	75.55	26.07	49.16	41.3	50.5	40.07	37.38	9.54	20.83	38.93
Model B	78.31	37.53	48.26	52.33	61.79	61.25	66.87	8.86	41.57	50.8

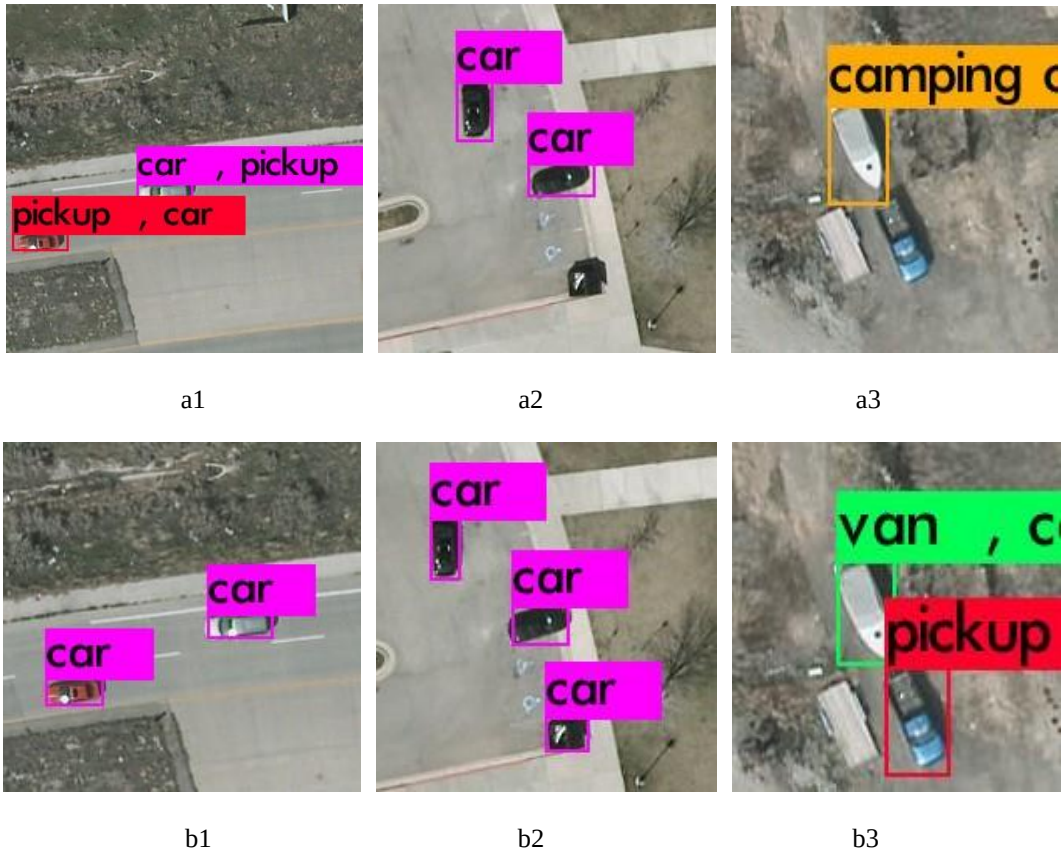
Proceedings DOI: 10.21467/proceedings.141
ISBN: 978-81-957605-4-1

Series: AIJR Proceedings
ISSN: 2582-3922

Vehicle detection parameters such as precision, recall, F-1 score and percentage of IoU on the VEDAI dataset using different YOLO versions and Model A and Model B are shown in Table 2. It was observed that when the percentage of mAP is high, the other parameters of object detection provided better results as well for particular vehicle detection algorithms. Model B provided better precision, recall, F-1 score and percentage of IoU than YOLOv4 Tiny and Model A 0.63, 0.57, 0.60 and 48.84 consecutively for Model B.

Table 2: *Vehicle recognition parameters of YOLO and developed models on VEDAI dataset*

Model	Precision	Recall	F-1 score	IoU (%)	mAP (%)
YOLOv3	0.7	0.71	0.69	54.45	61.79
YOLOv4	0.71	0.77	0.74	56.58	73.38
Yolov4-tiny	0.53	0.54	0.54	40.51	47.8
Model A	0.56	0.5	0.53	40.27	38.93
Model B	0.63	0.57	0.60	48.84	50.8



a1 a2 a3

b1 b2 b3

Figure 2: *Test image results of VEDAI dataset on using YOLOv4 Tiny (a1, a2, a3) and Model B (b1, b2, b3)*

Figure 2 shows the test image results using YOLOv4 Tiny in the first row and Model B in the second column. Since Model B provided better performance in recognising vehicles in this study, Model B and YOLOv4 Tiny test image results are shown in this section for comparison. YOLOv4 Tiny provided the wrong detection result in a1, and Model B provided the correct detected vehicle results in the b1 image. In images a2 and a3, every vehicle was not detected using YOLOv4 Tiny algorithm. However, Model B

detected all the vehicles in b2 and b3 images. Model B provided better average precision and other object detection parameters to recognise vehicles on the VEDAI dataset. Therefore, Model B is suitable for vehicle recognition systems to detect vehicles for autonomous transportation systems and traffic surveillance.

5 Conclusions

In this study, two lightweight convolutional neural network models based on YOLOv4 Tiny are developed to detect vehicles in aerial imagery. Model B provided better detection performance results than Model A because of the architecture's output size of the prediction boxes. The additional layers for the third prediction box increased the computational complexity but would also increase the mean average precision in Model B. To detect the small objects in the test images, the output image size in the second and third feature scale maps was increased in Model B. The weight files of the developed models increased after the training process due to the additional prediction box. In the future, Model B can be modified to reduce the computational complexity of the model and increase its accuracy and precision.

6 Declarations

6.1 Funding Source

The research funding was provided by Industry-Driven Innovation Grant (IDIG) by Universiti Malaya with project number PPSI-2020-CLUSTER-SD01.

6.2 Competing Interests

There is no conflict of interest.

6.3 Publisher's Note

AIJR remains neutral with regard to jurisdiction claims in published maps and institutional affiliations.

References

[1] S. Cepni, M. E. Atik, and Z. Duran, "Vehicle Detection Using Different Deep Learning Algorithms from Image Sequence," *Baltic Journal of Modern Computing*, vol. 8, no. 2, pp. 347-358, 2020. https://doi.org/10.22364/bjmc.2020.8.2.10

[2] Z. Yang and L. S. Pun-Cheng, "Vehicle detection in intelligent transportation systems and its applications under varying environments: A review," *Image and Vision Computing*, vol. 69, pp. 143-154, 2018. https://doi.org/10.1016/j.imavis.2017.09.008

[3] X. Zhang, H. Gao, C. Xue, J. Zhao, and Y. Liu, "Real-time vehicle detection and tracking using improved histogram of gradient features and Kalman filters," *International Journal of Advanced Robotic Systems*, vol. 15, no. 1, p. 1729881417749949, 2018. https://doi.org/10.1177/1729881417749949

[4] A. BOUGUETTAYA, A. KECHIDA, and A. M. TABERKIT, "A survey on lightweight CNN-based object detection algorithms for platforms with limited computational resources," *International Journal of Informatics and Applied Mathematics*, vol. 2, no. 2, pp. 28-44, 2019. Online access on June 10 2022 at https://dergipark.org.tr/tr/download/article-file/964257

[5] D. Xu and Y. Wu, "FE-YOLO: a feature enhancement network for remote sensing target detection," *Remote Sensing*, vol. 13, no. 7, p. 1311, 2021. https://doi.org/10.3390/rs13071311

[6] D. Zhu, G. Xu, J. Zhou, E. Di, and M. Li, "Object Detection in Complex Road Scenarios: Improved YOLOv4-Tiny Algorithm," in *2021 2nd Information Communication Technologies Conference (ICTC)*, 2021, pp. 75-80. https://doi.org/10.1109/ictc51749.2021.9441643

[7] P. Salavati and H. M. Mohammadi, "Obstacle detection using GoogleNet," in 2018 8th International Conference on Computer and Knowledge Engineering (ICCKE), 2018, pp. 326-332. https://doi.org/10.1109/iccke.2018.8566315

[8] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779-788. https://doi.org/10.1109/cvpr.2016.91

[9] H. V. Koay, J. H. Chuah, C.-O. Chow, Y.-L. Chang, and K. K. Yong, "YOLO-RTUAV: Towards Real-Time Vehicle Detection through Aerial Images with Low-Cost Edge Devices," *Remote Sensing*, vol. 13, no. 21, p. 4196, 2021. https://doi.org/10.3390/rs13214196

[10] W. Liu *et al.*, "Ssd: Single shot multibox detector," in *European conference on computer vision*, 2016, pp. 21-37.

[11] Online access on 10 June 2022 at https://link.springer.com/chapter/10.1007/978-3-319-46448-0_2

[12] T. Wang, R. M. Anwer, H. Cholakkal, F. S. Khan, Y. Pang, and L. Shao, "Learning rich features at high-speed for single-shot object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1971-1980. https://doi.org/10.1109/iccv.2019.00206

[13] S. Razakarivony and F. Jurie, "Vehicle detection in aerial imagery: A small target detection benchmark," *Journal of Visual Communication Image Representation*, vol. 34, pp. 187-203, 2016. https://doi.org/10.1016/j.jvcir.2015.11.002

[14] M. Ju, J. Luo, P. Zhang, M. He, and H. Luo, "A simple and efficient network for small target detection," *IEEE Access*, vol. 7, pp. 85771-85781, 2019. https://doi.org/10.1109/access.2019.2924960

[15] C. Chen, J. Zhong, and Y. Tan, "Multiple-oriented and small object detection with convolutional neural networks for aerial image," *Remote Sensing*, vol. 11, no. 18, p. 2176, 2019. https://doi.org/10.3390/rs11182176

[16] A. Amudhan and A. Sudheer, "Lightweight and computationally faster Hypermetropic Convolutional Neural Network for small size object detection," *Image Vision Computing*, vol. 119, p. 104396, 2022. https://doi.org/10.1016/j.imavis.2022.104396

[17] J. Gotthans, T. Gotthans, and R. Marsalek, "Prediction of Object Position from Aerial Images Utilising Neural Networks," in *2021 31st International Conference Radioelektronika (RADIOELEKTRONIKA)*, 2021, pp. 1-5. https://doi.org/10.1109/radioelektronika52220.2021.9420193

[18] Z. Jiang, L. Zhao, S. Li, and Y. Jia, "Real-time object detection method based on improved YOLOv4-tiny," *arXiv preprint arXiv:.04244*, 2020. Online access on 10 June 2022 at https://arxiv.org/abs/2011.04244