

PERFORMANCE ANALYSIS OF OBJECT DETECTION MODELS FOR VEHICLE-RELATED IMAGE SERVICES

JUN-HYUNG KO, NAMGI KIM¹

¹Department of Computer Science and Engineering, Kyonggi University, Suwon 16277

E-mail: ngkim@kgu.ac.kr

*Corresponding Author: Namgi Kim (ngkim@kgu.ac.kr)

ABSTRACT

Object detection is an actively researched field of computer vision, and notable research outcomes have been presented through an integration with deep learning. However, most previous studies on object detection have focused on evaluating the object detection performance for multiple classes. To a practical extent, such detection contrasts with how the type of classification required for object detection models is limited to a few numbers of classes. For example, the object detection classes required for autonomous vehicles or in vehicle detection services are limited to small specific classes, such as vehicles, persons, and road signs. In other words, the need has arisen to confirm which model exhibits an excellent performance for a small, specialized class such as vehicle object detection. Therefore, we evaluate representative object detection models to identify which models is more appropriate for vehicle object detection services. The results show that CenterNet [9] achieves the best performance for vehicle object detection during autonomous driving and for CCTV use among the three models, followed by YOLOv4 [7] and SSD [8].

Keywords: *Object detection; Image processing; Deep learning*

1. INTRODUCTION

Object detection has long been studied in the fields of computer vision and image processing. AlexNet [1], which emerged in 2012, has significantly improved the object detection performance by using deep learning technology based on a convolutional neural network, which later prompted deep learning technology to be widely applied in the field of object detection.

Since the emergence of AlexNet, object detection using deep learning technology has focused on the detection of multiple objects in various classes. The MS COCO [2] data set, which has frequently been used for measuring the object detection performance, has 80 different object classes. The object detection performance for multiple classes is certainly a critical factor in identifying the general performance of a specific model. However, practical services mostly do not require various objects to be classified at once, and instead, certain objects specialized for the services need to be detected intensively.

One of the fields in which object detection can be positively utilized is autonomous vehicles. Object detection is one of the diverse technologies employed by autonomous vehicles. If object detection can be successfully conducted through

images, the precision of vehicle position detection can be improved through the use of simultaneous localization and mapping (SLAM) when detecting front and rear vehicles. Tesla CEO Elon Musk proposed that object detection through deep learning is more crucial for establishing autonomous vehicle technology than the conventional SLAM technique [3].

Important elements of object detection in autonomous vehicles include vehicles and humans. Specifically, when a deep learning model is applied in autonomous vehicles, the model does not need to detect all 80 classes including chairs, airplanes, and balloons. However, previous studies have applied object detection for various classes and the results were comparatively analyzed. Although such a comparative analysis is appropriate for identifying the overall performance of a model, it is inappropriate for assessing whether the model corresponds to a specific purpose. Therefore, the ability to detect a specific small class of deep learning models must be verified.

This study thus evaluates and comparatively analyzes the performance of various object detection models for the goal of a vehicle object detection service, which is one of the areas with a high practical potential in the object detection field.

The MS COCO data set was used for a comparative analysis, and a small-class dataset targeting vehicle types in this particular dataset was used for vehicle object detection. Two-stage detectors, YOLOv4 [7], SSD [8], and CenterNet [9], which are appropriate for a practical use environment, were selected for comparison to fit the purpose of vehicle object detection. In this paper, we just evaluate the previous object detection models based on deep learning and did not propose a new model.

The rest of this paper is organized as follows. In Chapter 2, previous studies on object detection are examined. In Chapter 3, the environment and detailed configuration for conducting object detection training are defined. In Chapter 4, the vehicle object detection performance is examined based on the actual training results of each selected model. In Chapter 5, the results are comparatively analyzed to select a model suitable for a specialized purpose, i.e., vehicle object detection. Finally, in Chapter 6, some concluding remarks and areas of future research are described.

2. RELATED WORK

Prior to model selection, object detection must be conducted in real time for vehicle object detection. In general, videos typically used in a control system that applies vehicle object detection such as CCTV have a minimum of 30 frames per second (FPS) [4][5]. Therefore, a model must be capable of processing at a rate of at least 30 FPS to conduct object detection in real time.

Since AlexNet first applied deep learning technology to object detection, most object detection models have been categorized as one- or two-stage models. Object detection consists of a regional proposal process for finding the scope and location in which objects may be located in an image and a classification process for classifying the objects detected using the region. The criteria for categorizing detection models are based on the differences in the way these processes are applied. Two-stage detection models involve regional proposal and classification being carried out in a sequential manner. By contrast, regional proposal and classification are simultaneously carried out in one-stage detection models. Owing to such nature, one-stage models have a faster processing speed than two-stage models [6]. The system considered in this study must be capable of processing at a rate of at least 30 FPS because objects must be processed from videos in real time. Therefore, one-stage detection models are appropriate for a system

that detect objects in real-time videos. This study analyzed the performance of three well-known object detection models among one-stage detectors.

YOLOv4 [7] is an enhanced version of a model series collectively referred to as YOLO. YOLOv4 aims for quick training using a general single GPU. YOLO v3, which is used as the infrastructure of YOLOv4, uses Darknet-53 as the backbone network and secures a fast inference speed based on a multi-scale feature map. Furthermore, YOLO predicts the location of an object through an anchor box during the regional proposal process. The performance was also improved by combining different techniques with a simple model. A Bag of Freebies (BoF) is a technique used for training a model to demonstrate a better performance without increasing the inference cost. Although a BoS slightly increases the inference time of a model, the performance can be effectively improved despite the consumption cost. Accordingly, YOLOv4 pursued an enhanced performance of the model by combining various approaches and techniques.

SSD [8] was proposed as an enhanced model of the first version of YOLO, YOLO v1. YOLO v1 considerably increased the object detection speed by carrying out an integrated regional proposal and classification. However, the accuracy of the model is decreased if a relatively smaller dataset is provided because the model only selects two border regions per grid cell. SSD applies a multi-scale feature map as a solution for such problems. It aims to solve the problem of a reduced responsiveness to various sizes in a single sized feature map. This technique was also applied in the aforementioned YOLOv4, in which the effectiveness has been proven.

CenterNet [9] is also a one-stage detection model similar to YOLO and SSD but with one noticeable difference. A conventional one-stage detector uses an anchor box during a regional proposal. Although the anchor box method is certainly an effective measure, a difference between the truth anchor box and an error anchor box may occur. An increase in this difference leads to a decreased training performance; thus, selecting an appropriate anchor box size is a crucial factor. CenterNet proposed a key point estimation method using one anchor point as an improvement measure for the anchor box method. Unlike how conventional models proceed with training based on the degree of overlap between anchor boxes, this model is trained using the center point location of an object as a probability. Conventional anchor

box models require the generation of various object border candidates where the borders with the highest probability are left through the process of non-maximum suppression (NMS) [10]. The key point estimation method of CenterNet eliminates the need of the NMS process, which ultimately increases the detection speed. In this study, three object detection models are trained with a small class object dataset suitable for vehicle object detection systems, and their detection performance is analyzed.

3. TRAINING AND EXPERIMENT METHOD

The PC environment used in the experiment is shown in Table 1, and three one-stage detection models, YOLOv4, SSD, and CenterNet, were trained. The MS COCO dataset was used for training. From the 2017 MS COCO dataset, three classes related to vehicles, i.e., cars, buses, and trucks, were extracted. The detailed configuration of the dataset is presented in Table 2. The trained model is applied to two types of vehicle videos having different forms and features. The video types include vehicle driving videos and CCTV videos, the features of which are shown in Table 3. The hyperparameter configuration of each model is as shown in Table 4.

Table 1: Training environment of object detection models

Parameters	Versions
OS	Ubuntu 16.04 LTS [11]
GPU	Tesla V100-SXM2-32GB * 2EA
Anaconda	Anaconda v.1.6.14 [12]
PyTorch	PyTorch v.1.4.0 [13]
CUDA	CUDA v.11.1(+11.2) [14]
Cudnn	Cudnn v.8.1.1 [15]

Table 2: Detailed configuration of training dataset

Parameters	Training	Validation
Car	11,261	3,864
Bus	1,906	570
Truck	3,579	830
Total	16,746	5,264

Table 3: Specific parameters of training dataset

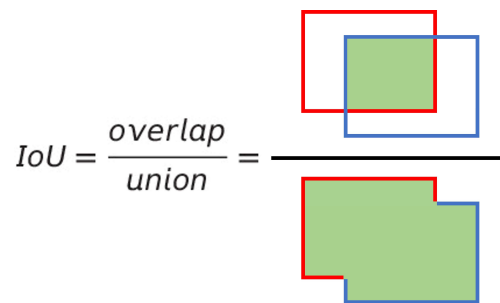
Image Type	Traffic	CCTV
Resolution	720p	1080p
FPS	30	60
Duration	5min	2min

Table 4: Hyper-parameters for models

	YOLOv4	SSD	CenterNet
Optimal function	Adam	Adam	Adam
Batch size	64	64	32
Learning rate	0.0005	0.0001	0.000125
Patch size	832x832	300x300	512x512

Three models were trained with a dataset composed of three classes, and their detection results were analyzed. In this section, the performance evaluation indices for object detection models are defined. The three elements used as indices include the sensitivity, accuracy, and detection speed.

The Intersection over Union (IoU) shown in Figure 1 is an index for evaluating the object detection performance and refers to the ratio of the ground truth to the intersection of the predicted border region [16][17]. The model is considered to have achieved a correct prediction if the ratio of this intersection is above the threshold. The threshold of the IoU was set to 50% and 75% to compare the performance of each model according to the threshold.



$$IoU = \frac{\text{overlap}}{\text{union}} = \frac{\text{green area}}{\text{red and blue area}}$$

Figure 1: Definition of Intersection over Union (IoU)

The average precision (AP) is an evaluation index used for measuring the object detection accuracy of a model [18]. Precision and recall are

the most important classification performance evaluation indices for identifying the detection performance of a model. The precision is also treated as a positive predictive value (PPV) and is the ratio of the actual true values among the values classified as true. The recall is treated as the sensitivity and is the ratio of the true values predicted by a model among the actual true values. The AP is an index for evaluating the performance of a model with a numerical figure by reflecting both the precision and recall. The mean AP (mAP) is the mean AP value calculated for all classes and is commonly used when discussing the overall performance of a model.

The speed for processing the object detection per second is expressed in the number of frames per second. For object detection to be practically applied in autonomous vehicles, a real-time processing capability is inevitably required. Therefore, to provide relevant services in autonomous vehicles in real time, the object detection processing speed must be at least 30 FPS. Accordingly, the object detection processing speed of each model is measured, and the performance of each model is analyzed and compared based on the measurement results.

4. EXPERIMENT RESULTS

4.1 YOLOv4

Figure 2 shows the graphs illustrating the performance of the YOLOv4 model. Table 5 shows the difference between the maximum and minimum values of the AP per class. The training was carried out for up to 150,000 iterations. Table 6 presents the deviation per class for the training amount at which the mAP exhibited the best performance per IoU threshold.

Table 5: Comparison of AP range of YOLOv4 according to training amount

IoU	50%	75%
Car	5.32%	7.90%
Bus	7.20%	12.28%
Truck	8.45%	16.75%
mAP	5.99%	12.31%

Table 6: AP Deviation of YOLOv4 with training iterations of maximum mAP

IoU	50%	75%
Training iterations	70,000	90,000
Car	0.57%	15.32%
Bus	3.27%	10.19%
Truck	3.83%	3.25%

The performance measurement results showed that the slope in the graph illustrating the fluctuation of AP consistently varied regardless of the increase in the training amount of YOLOv4. In particular, mAP decreased even further at the end of the training compared to the beginning of the training. It can be concluded that an increase in the training amount does not guarantee the stability in the performance of YOLOv4. In Table 5, the deviation per class in YOLOv4 is within 4% at an IoU of 50%. It was confirmed that the focal loss [19] applied to the model partially resolves the class imbalance problem of the dataset. However, the index demonstrates a completely opposite trend at an IoU of 75%. When the index values at an IoU of 75% in Table 5 were analyzed, the mAP range recorded a difference of 12.31%, whereas the range of the truck class was 16.75%. The deviation at an IoU of 75% in Table 6 showed that the truck class did not show a significant difference from the deviation at an IoU of 50%. However, the car and bus classes demonstrated a significant difference of 15.32% and 10.91%, respectively. It can thus be concluded that the performance of YOLOv4 is substantially degraded at an IoU of 75% compared to an IoU of 50%. At an IoU of 75%, the trend of resolving the dataset imbalance through the focal loss also differs. Compared to a 3.25% deviation of the truck class, the deviation of the car class is 15.32%, resulting in a difference of 12.07%. This difference is significantly large compared to the 3.26% difference between the car and truck classes at an IoU of 50%.

When the performance of YOLOv4 was evaluated, the model was deemed to be capable of achieving object detection at an IoU of 50% even when a dataset with a class imbalance is used. However, the performance considerably degrades at an IoU threshold of 75%.

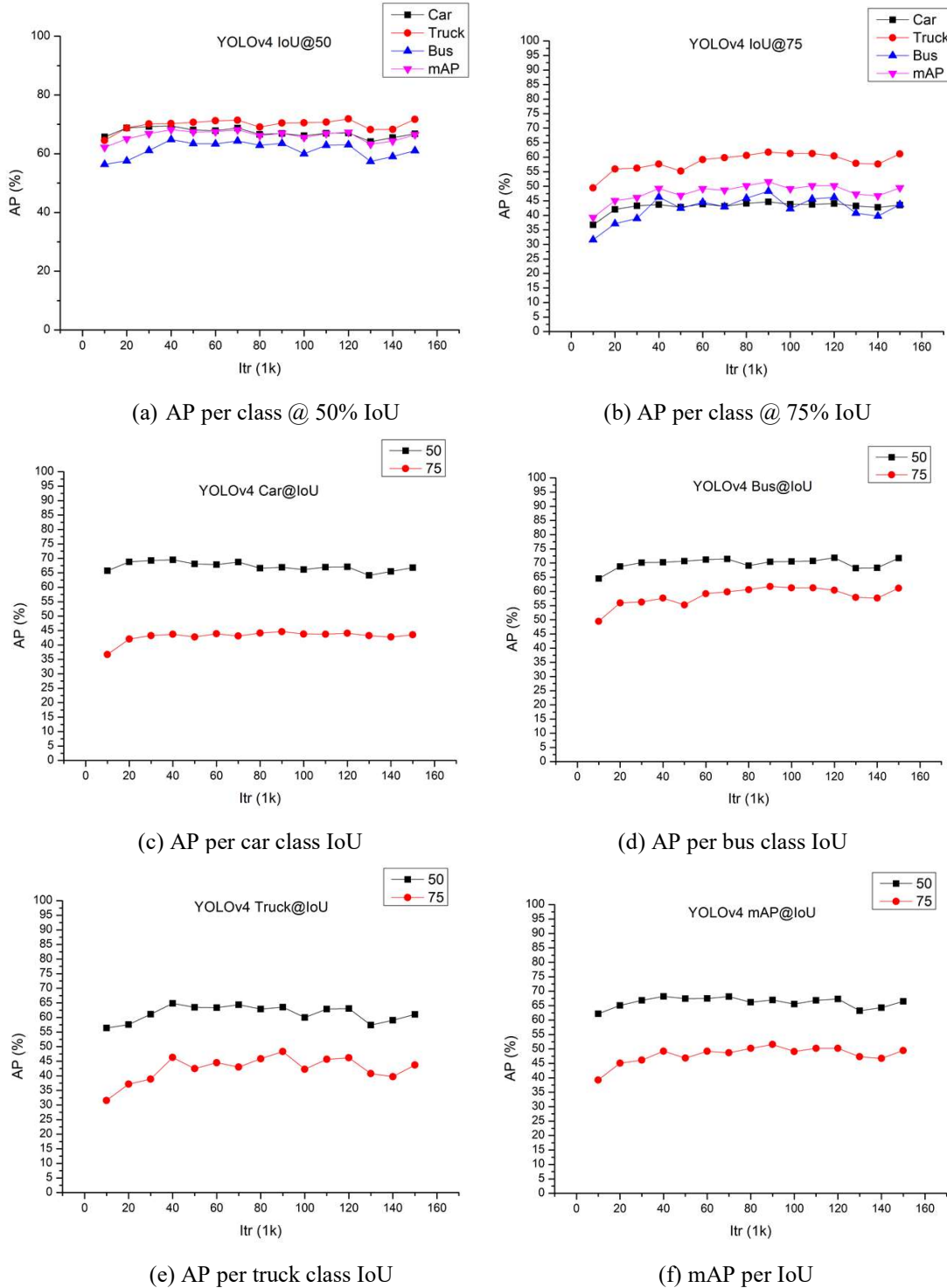


Figure 2: Performance of YOLOv4

4.2 SSD

Figures 3 shows graphs representing the performance of the SSD model. Table 7 shows the

difference between the maximum and minimum AP values per class. Table 8 shows the deviation per class at the point where mAP showed the best

performance for each IoU threshold value. The model was trained for up to 200 epochs.

SSD demonstrated a more stable performance compared to YOLOv4. The mAP performance value fluctuates within 2% after the initial training has been carried out and gradually increases.

Compared to YOLO, however, the class imbalance problem worsened in the SSD. As shown in Table 8, the deviations of the car and truck classes are 15.9% and 7.7%, respectively, and the difference in the AP value between the two classes is 23.6%. Such a figure is significantly large when considering that the deviation between classes was

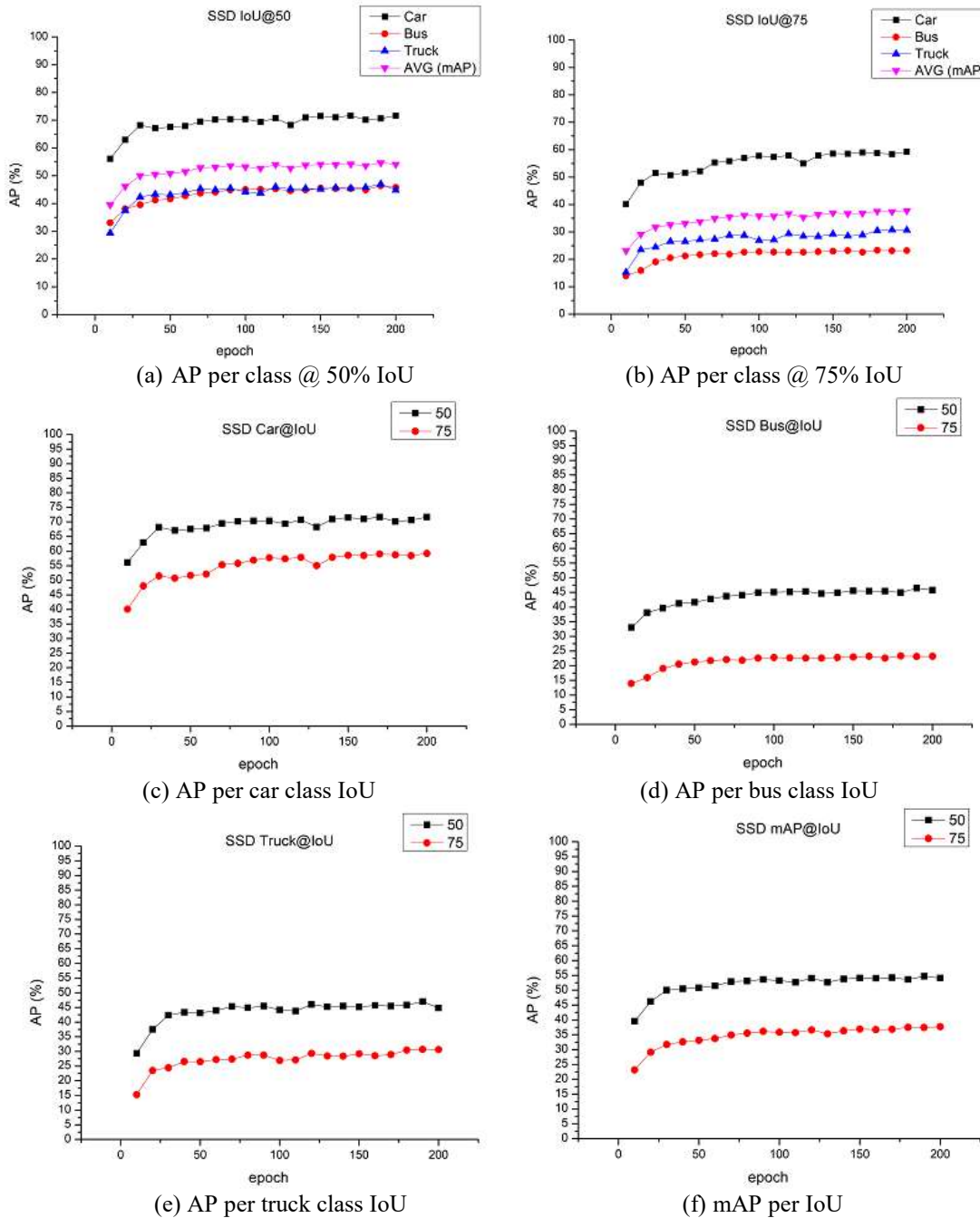


Figure 3: Performance of SSD

within 4% in the YOLOv4 model. A similar phenomenon is also observed at an IoU of 75%. Unlike YOLOv4, where the focal loss is applied to resolve the issue of a class imbalance, no measure was taken in this model, which resulted in a class imbalance of the dataset being accurately reflected. Considering this limitation, the AP value of the car class with the largest dataset is higher than that in YOLOv4 at IoU 50%. The difference between classes is even greater at an IoU of 75% compared to an IoU of 50% owing to a data imbalance. According to the result of analyzing the deviation between classes in Tables 7 and 8, the deviation of the car and bus classes is 21.5% and 14.54%, respectively. The difference between the two classes is 36.04%, which is considerably large. Compared to YOLOv4, SSD is a model in which the performance stability is guaranteed depending on the training amount but still entails a limitation in that the imbalance in the dataset cannot be resolved.

Table 7: Comparison of AP range of SSD according to training amount

IoU	50%	75%
Car	15.62%	19.13%
Bus	13.35%	9.36%
Truck	16.64%	15.40%
mAP	15.20%	14.60%

Table 8: AP Deviation of SSD with training iterations of maximum mAP

IoU	50%	75%
Training epochs	190	200
Car	15.9%	21.5%
Bus	8.30%	14.54%
Truck	7.70%	7.07%

4.3 CenterNet

Figure 4 shows the AP graph per class in the CenterNet model at an IoU of 50% and an IoU of 75%, respectively. Table 9 shows the difference between the maximum and minimum values of the AP per class. Table 10 presents the deviation per class at the training amount at which the mAP exhibited the best performance per IoU threshold. The training was carried out for up to 200 epochs.

Similar to YOLOv4, CenterNet is also applied with a focal loss to partially resolve the problem of a class imbalance in the dataset. However, the problem was resolved in a different way from YOLOv4. In CenterNet, the class with the highest AP value was the bus class rather than the car class. In Table 10, the deviation of the car class at an IoU of 50% and an IoU of 75% is 2% and 6%, respectively, whereas the deviation of the bus class is 15.5% and 22.8%, respectively. The difference in the AP values between the bus class and the truck class at an IoU of 50% is 20.1%. These results indicate that the imbalance in the dataset was not sufficiently resolved even when the focal loss was applied, unlike with YOLOv4. CenterNet achieved the optimal performance with a small amount of training, i.e., 40 epochs, unlike the other two models, and further training did not improve the performance. Instead, the performance was stabilized as the fluctuation in the AP values was minimized. Overall, CenterNet has the advantage of requiring less training to reach the optimal performance level compared to two previous models. Similar to YOLOv4, the problem of a class imbalance in the dataset was partially resolved by applying the focal loss, but to a limited extent.

Table 9: Comparison of AP range of CenterNet according to training amount

IoU	50%	75%
Car	10.0%	14.9%
Bus	16.3%	24.1%
Truck	20.2%	17.1%
mAP	16.2%	18.8%

Table 10: AP Deviation of CenterNet with training iterations of maximum mAP

IoU	50%	75%
Training epochs	50	50
Car	2%	6%
Bus	15.5%	22.8%
Truck	4.6%	10.1%

4.4 Comparison

Figure 5 shows the mAP of each model at an IoU of the 50%. As shown in the results, SSD, which was the earliest released, achieves the poorest performance among the one-stage detectors. Furthermore, SSD does not overcome the class

imbalance in the dataset. Contrarily, YOLOv4 and CenterNet are applied using techniques for overcoming the class imbalance, thus resulting in a better performance. In particular, YOLOv4 exhibited the best performance among the two

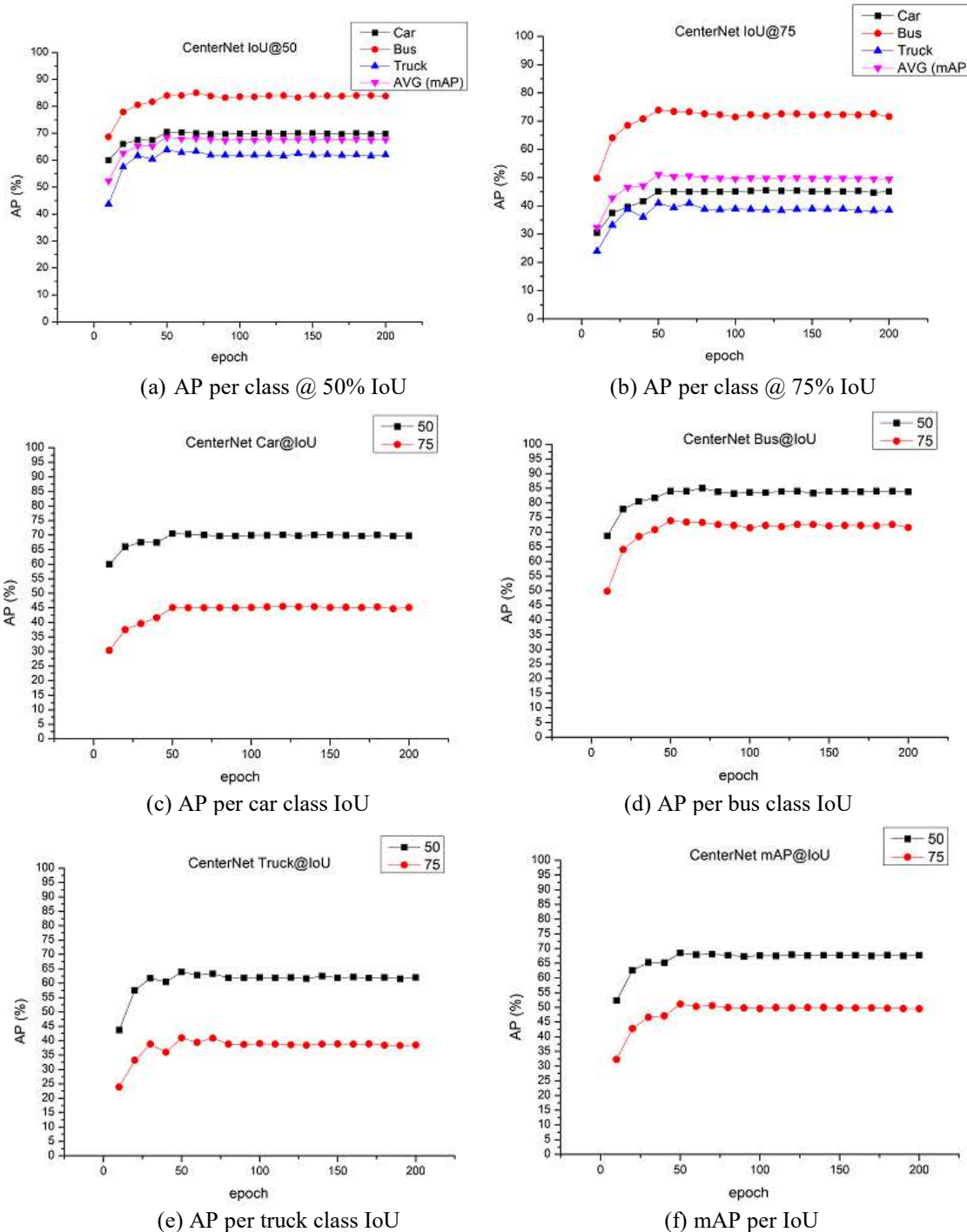


Figure 4: Performance of CenterNet

models by effectively resolving the class imbalance.

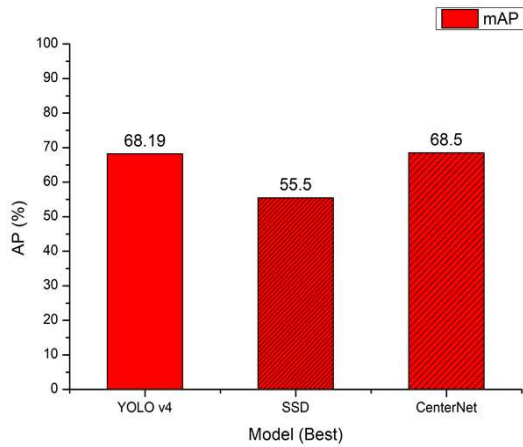


Figure 5: mAP at 50% IoU

When vehicle object detection becomes commercialized, one of the important factors required will be the real-time processing speed. In this study, the FPS of the models was compared for two types of video format, i.e., autonomous vehicle and CCTV for-mats. The detailed features of each video format are presented in Table 3.

Figure 6 shows a graph of the processing speed when each model applied real-time object detection in two video formats. YOLOv4 recorded 15 FPS for both video types, whereas SSD recorded 5 FPS for both video types. Only CenterNet recorded 30 FPS or higher, which is the standard for real-time video processing. Specifically, CenterNet showed 30 FPS for vehicle driving videos and 23.4 FPS for CCTV videos. The difference in processing speed between the two video types possibly occurred because 1) the resolution of the vehicle driving videos is 720p whereas that of CCTV videos is 1,080p, and 2) there is a difference in the number and size of the objects being detected simultaneously

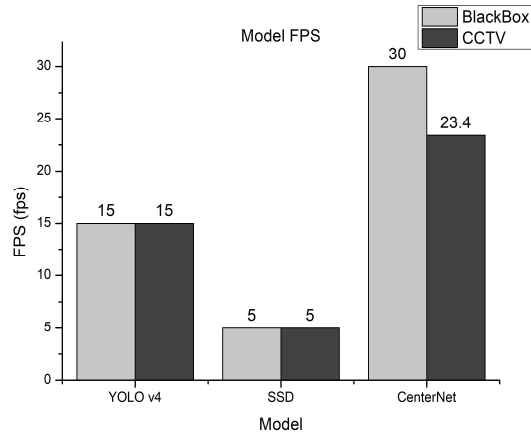


Figure 6: Object detection speed according to the video types

5. ANALYSIS

YOLOv4 has the smallest deviation between classes among the three models. The difference between AP and mAP of the bus class at an IoU of 50% is 3.37% which was the lowest. It can thus be concluded that YOLOv4 can partly resolve the issue of an imbalance even when a dataset with a class imbalance is used. Furthermore, an increase in the training amount did not guarantee the stability of the performance in comparison to the other models. The video processing speed was 15 FPS, which is significantly less than 30 FPS, or the speed required for real-time processing.

SSD, the oldest of the three models, achieved a low performance level overall. Among the three models, SSD also had the greatest deviation in AP values with a maximum difference of 36% between classes, thus failing to resolve the issue of a class imbalance. In addition, the video processing speed among the three models was the slowest at 5 FPS.

CenterNet demonstrated the fastest training achieved among the three models by exhibiting the optimal performance even after 45 epochs. Similar to YOLOv4, the issue of an imbalance was partially resolved despite using a dataset with a class imbalance for training. However, the results were contrasted with those of YOLOv4. An increase in the training amount did not improve the performance, unlike with YOLOv4. The most noticeable feature of CenterNet is the real-time responsiveness in which CenterNet was the only model that exceeded the video processing speed required for real-time services, or 30 FPS, during the experiment.

6. CONCLUSIONS

Despite using a small class dataset with a data imbalance, YOLOv4 still recorded the smallest deviation between classes among the compared models. The mAP index was at least 60% overall at an IoU of 50%. The value was fairly low at an IoU of 75%, but was sufficient for the size of the objects used in vehicle object detection videos for autonomous vehicles. YOLOv4 achieved only 50% of the required detection speed, or 30 FPS. If the model is lightened or training is initially applied with a lightweight version, the detection speed may be higher than 30 FPS, enabling real-time processing.

CenterNet was the only model in this study that demonstrated an image processing speed of 30 FPS or higher for responding to real-time image processing of autonomous vehicles. AP in the bus class was approximately 70% at an IoU of 75%. This contrasts with the image processing speed of YOLOv4, i.e., 15 FPS, and the AP value at an IoU of 75% does not exceed 60%. However, CenterNet failed to achieve a sufficient level of performance similar to that of YOLOv4 even when the problem of a class imbalance was attempted to be resolved through the use of the focal loss. CenterNet may outperform YOLOv4 in terms of detection accuracy if a class imbalance is not present in the dataset.

SSD, which is the oldest of the three models, does not have a sufficient level of performance required for services used in autonomous vehicles overall. Ultimately, CenterNet can be considered the most appropriate for vehicle object detection in actual services required for autonomous vehicles if the dataset does not have a class imbalance.

In this paper, we analyzed the performance of object detection models based on the deep learning technique for vehicle-related services. In the paper, we compare the performance of the representative object detection models such as YOLOv4, CenterNet, and SSD. However, we did not propose a new object detection model.

Future research must be focused on determining whether the performances of YOLOv4 and CenterNet can be reversed if the dataset imbalance is resolved, and on selecting the optimal parameters for vehicle object detection for autonomous vehicle services. If the problem of a data imbalance cannot be resolved, other data preprocessing techniques will be studied as well. Finally, the performance of two-stage detectors will be compared to analyze

whether such detection models can also be used for real-time services.

ACKNOWLEDGMENTS:

This work was supported by Kyonggi University Research Grant 2021.

REFERENCES:

- [1] A. Krizhevsky, I. Sutskever, G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", NIPS 2012.
- [2] MS COCO dataset, Available online: <https://cocodataset.org/#home> (accessed on 5 January 2021).
- [3] TechTalks, "Why deep learning won't give us level 5 self-driving cars", Available online: <https://bdtechtalks.com/2020/07/29/self-driving-tesla-car-deep-learning/> (accessed on 5 January 2021).
- [4] Wikipedia, "Real-time computer graphics", Available online: https://en.wikipedia.org/wiki/Real-time_computer_graphics (accessed on 5 January 2021).
- [5] Meakyung Media Group, <https://www.mk.co.kr/news/business/view/2019/03/146590/> (accessed on 5 January 2021).
- [6] S. K. Han, S. C. Kwon, "Real-time object detector for small object detection in the video", Proceedings of Symposium of the Korean Institute of Communications and Information Sciences, 2019, pp. 201-202.
- [7] A. Bochkovskiy, C.-Y. Wang, H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection", arXiv:2004.10934, 2020.
- [8] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A. C. Berg, "SSD: Single shot multibox detector", arXiv:1512.02325, 2015.
- [9] X. Zhou, D. Wang, P. Krhenbhl, "Objects as points", arXiv:1904.07850, 2019.
- [10] LearOpenCv, "Non maximum suppression: Theory and implementation in PyTorch", Available online: <https://learnopencv.com/non-maximum-suppression-theory-and-implementation-in-pytorch/> (accessed on 5 January 2021).
- [11] Ubuntu, <https://ubuntu.com/> (accessed on 5 January 2021).
- [12] Anaconda, <https://www.anaconda.com/> (accessed on 5 January 2021).
- [13] PyTorch, <https://pytorch.org/> (accessed on 5 January 2021).

-
- [14] CUDA, <https://developer.nvidia.com/cuda-toolkit> (accessed on 5 January 2021).
 - [15] cuDNN, <https://developer.nvidia.com/cudnn> (accessed on 5 January 2021).
 - [16] Pyimagesearch, "Intersection over Union (IoU)", <https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/> (accessed on 5 January 2021).
 - [17] C3.ai, "GroundTruth", <https://c3.ai/glossary/machin-learning/ground-truth/> (accessed on 5 January 2021).
 - [18] Wikipedia, "Evaluation measures (information retrieval)", [https://en.wikipeida.org/wiki/Evaluation_measures_\(information_retrieval\)#Mean_average_precision](https://en.wikipeida.org/wiki/Evaluation_measures_(information_retrieval)#Mean_average_precision) (accessed on 5 January 2021).
 - [19] Facebook Research, "Focal loss for dense object detection", <https://research.fb.com/publications/focal-loss-for-dense-object-detection/> (accessed on 5 January 2021).