香 港 大 學

THE UNIVERSITY OF HONG KONG

The University of Hong Kong

Department of Computer Science

Computer Vision: Object Detection for Autonomous Driving

Author:
CAO Kepan

UID:
3036032059

An Assignment submitted for the HKU:

DASC7606 Deep Learning

March 9, 2023

1 Introduction

Object detection plays a vital role in enabling autonomous driving. In this report, we present a detailed analysis of several empirical experiments, where we explore the outcomes of varying parameter settings on the performance of the object detection model. Additionally, we delve into the fundamental principles of deep learning to gain insights into the effectiveness of network architectures such as ResNet and YOLO v1.

To evaluate the model's performance, we employ the mean average precision (mAP) metric, which provides a comprehensive measure of the model's accuracy across multiple experiments and parameter settings. Our objective is to gain a comprehensive understanding of the performance of the object detection model under different experimental conditions and parameter settings.

2 Improvement for Higher mAP

2.1 Initial Parameters

Initial Parameters	
Number of CUDA Devices	1
Batch Size	8
Learning Rate	1e-5
Epoch	10

We have utilized this specific branch of parameters to train our model, and the results of the loss and validation loss have been illustrated below:

However, upon evaluating the performance metrics, we have observed that the mAP of the training model is relatively low. Moreover, we have observed that the validation loss exhibits a tendency to converge towards 5 after six epochs of training. We recognize that the current training iteration is insufficient to obtain the optimal model, thus necessitating a strategic modification to the training methodology. As such, we have decided to increase the epoch and modify the learning rate to 5e-5, with the aim of achieving an improved model performance.

2.2 Improvement

Upon conducting the preliminary training process, our observations have revealed that the mAP metric of the model is not attaining the desirable range. In light of this, we have deduced that a vital step in the subsequent training procedure would be to undertake parameter tuning. By doing so, we aim to enhance the model's performance and optimize its overall accuracy

Class	AP
Pedestrian	2.15
Cyclist	18.01
Car	50.03
Truck	16.08
Tram	0.85
mAP: 0.17	

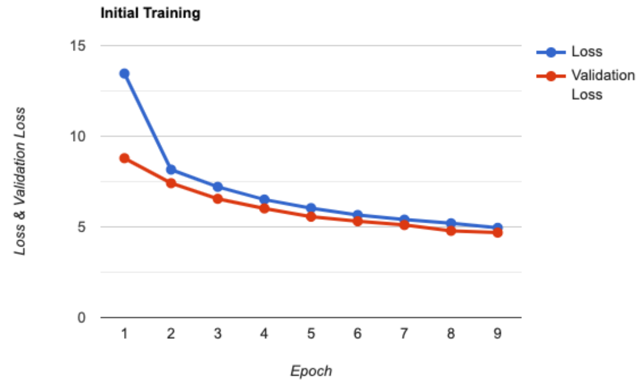

Table 1: Initial Training

Improvement 1	
Number of CUDA Devices	1
Batch Size	8
Learning Rate	5e-5
Epoch	20

Ascertaining the optimal parameters for training is a critical aspect in determining the efficacy of the training process. In this context, the epoch serves as a pivotal factor in determining the training effectiveness. In view of this, we have undertaken an approach to enhance this parameter by increasing it to a value of 20. Subsequently, we present the loss curve and mAP table, providing an insightful analysis of the training process and its corresponding performance metrics.

Class	AP
Pedestrian	16.95
Cyclist	33.44
Car	61.27
Truck	44.28
Tram	29.62
mAP: 0.37	

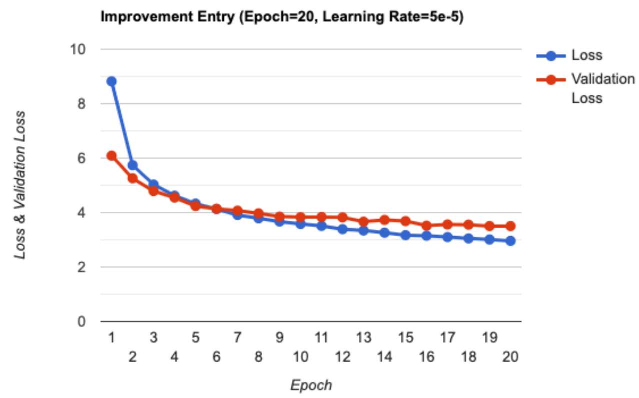

Table 2: Improvement 1

Given the prevailing conditions, we have determined that augmenting the number of epochs yields a significant enhancement in the mAP table. We have ascertained this through our analysis of the loss curve, which revealed a discernible convergence trend in validation loss to a value of 3.5 after conducting 17 epochs. This convergence exhibited a substantial decrease of approximately 25%.

Following the deliberation of modifying the number of epochs, we proceeded with several additional training sessions employing epoch values of 30 and 50, respectively.

Our findings indicate that continuing to augment the epoch count results in a further reduction in validation loss, as evidenced by the monotonic convergence trend. However, we have observed that despite the continued decrease in validation loss, the model's performance in terms of mAP would not be significantly improved. This phenomenon may be attributable to overfitting issues and the constraints of image size input. Given the marginal difference in mAP performance between epoch values of 30 and 50, as observed in real training results, we have decided to skip epoch=50 and provide the loss curve and mAP for epoch=30.

Class	AP
Pedestrian	19.61
Cyclist	34.78
Car	64.16
Truck	43.89
Tram	32.99
mAP: 0.39	

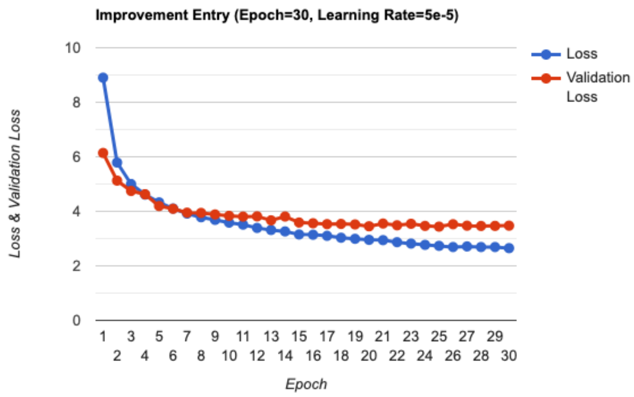

Table 3: Improvement 2

3 Results and Discussion

3.1 Learning Rate Decay (lrDecay)

In light of the popularity of the lrDecay technique in training modern neural networks, it is commonly believed that an initially large learning rate accelerates training or helps the network escape spurious local minima, and decaying the learning rate helps the network converge to a local minimum and avoid oscillation. However, recent experiments suggest that these beliefs are insufficient in explaining the general effectiveness of lrDecay in training deep, wide, and nonconvex neural networks.

Based on the research presented in the past research paper, the common beliefs about how lrDecay works in modern neural networks are insufficient in explaining its effectiveness (Cai et al., 2020). While the optimization analysis of (Stochastic) Gradient Descent provides some explanation, the authors argue that the complexity of modern neural networks requires a more comprehensive understanding. The experiments conducted in the paper demonstrate that lrDecay is effective in training modern neural networks such as WideResNet, which is deep, wide, nonconvex, and suitable for datasets like CIFAR10 (Krizhevsky & Hinton, 2009) (Cai et al., 2020). The authors propose a novel explanation for the effectiveness of lrDecay: the effect of a decaying learning rate is to improve the learning of complex patterns, and the effect of an initially large learning rate is to avoid memorization of noisy data. It is supported by experiments on a dataset with tractable pattern complexity as well as on real-world datasets

Followed by the findings and insights provided by recent research on lrDecay, we have decided to refrain from incorporating this technique in our assignment. Instead, we will focus on the simple yet effective parameter-tuning strategies of adjusting the number of epochs and learning rate to optimize the performance of our model. This decision is guided by the aim to avoid potential overfitting issues and to ensure the reproducibility of our results by using a straightforward approach in our experimentation.

3.2 Visualisation of model predictions

Based on our prior attempts, we have achieved the optimal mAP value of 0.39 by approaching a total of 30 epochs, a notable improvement compared to the same mAP value obtained with shorter training times of 50 epochs. To assess the effectiveness of our model predictions, we generated a total of 20 pictures, from which we have selected two for presentation in this report.

We found that the precision of our model in detecting pedestrians is relatively low, as suggested by the low confidence values assigned to the black boxes. In contrast, our model demonstrated significant proficiency in identifying cars, as reflected by the high confidence values attached to the green boxes. However, upon reviewing the image below, we noted an excessive number of boxes, which could be indicative of model overfitting.

These results indicate the potential of deep learning techniques for object detection in images. Future research could focus on improving the precision of our model in detecting pedestrians and exploring strategies to mitigate the impact of overfitting on our model performance.

4 Conclusion

The report outlines several efforts that have been made to improve the detection performance of the object detection model.

Firstly, we observed that the model's mAP was relatively low under the initial parameters, and the validation loss was converging towards 5 after six epochs of training. To address this issue, we decided to increase the number of epochs and modify the learning rate to 5e-5 to achieve better model performance.

Secondly, it is recognized that parameter tuning is critical in enhancing the model's performance and optimizing its overall accuracy. We increased the epoch count to 20 and conducted an analysis of the loss curve and mAP table to gain insights into the training process's performance metrics. Also, we observed that augmenting the number of epochs significantly improved the mAP table, with a substantial decrease of approximately 25% in the validation loss.

Finally, we proceeded with several additional training sessions using epoch values of 30 and 50, respectively. While continuing to augment the epoch count resulted in a further reduction in validation loss, we observed that the model's performance in terms of mAP would not be significantly improved beyond epoch=30. This phenomenon may be attributable to overfitting issues and the constraints of image size input. Therefore, we decided to skip epoch=50 and provide the loss curve and mAP for epoch=30.

Overall, we employed a systematic approach of modifying the number of epochs and learning rate and conducting an analysis of the performance metrics to improve the detection performance of the object detection model.