
Object Detection을 활용한 자율주행 보조장치 구현

김서영
Aiffel
seoyeongkim09@gmail.com

Abstract

자율주행 분야에서 Object Detection은 속도와 정확성을 포함한 여러 요소들이 중요한 역할을 한다. 본 연구에서는 RetinaNet 모델을 구현하고, 이를 기반으로 자율주행 보조장치를 개발하였다. 실험을 통해 해당 모델의 최적화 시점을 시각화하여 비교해 본 후, 각 시점이 객체 인식 성능에 미치는 영향을 살펴보고 자율주행 보조장치의 성능을 평가하였다. 이 연구는 자율주행 기술에 있어 Object Detection의 적합성에 대해 논의하고자 한다.

1 Introduction

본 연구는 Object Detection이 자율주행 분야에서 어떻게 적용될 수 있는지, 그리고 그 활용 가능성에 대해 분석하였다. Object Detection은 크게 Single stage detector와 Two stage detector로 나뉜다. Two stage detector는 슬라이딩 윈도우(Sliding Window) 방식을 사용하며, 정확도 면에서는 유리하지만 연산량이 많고 속도가 느린 단점이 있다. 반면 Single stage detector는 연산량이 적어 실시간 처리에 유리하지만, 정확도에서는 상대적으로 낮을 수 있다. 자율주행 분야에서는 사용자의 안전을 최우선으로 고려해야 하므로, 속도와 정확성 간의 균형을 중시한다. 이에 본 연구에서는 RetinaNet을 선택하여 이 모델이 자율주행 보조장치에 적합한지 분석하였다.

2 Related works

Single stage detector는 YOLO와 SSD가 있으며, Two stage detector는 RCNN과 Faster RCNN이 포함된다. Single stage detector는 속도가 빠르지만, 정확도에서는 Two stage detector에 비해 떨어지는 경향이 있다. 자율주행 시스템은 실시간 처리와 정확성을 동시에 만족해야 하므로, 이 두 가지 요소를 고려할 수 있는 모델이 필요하다. RetinaNet은 FPN(Feature Pyramid Network)과 Focal Loss를 활용하여 이러한 문제를 해결하고자 하였다. FPN은 다양한 크기의 객체를 탐지할 수 있도록 멀티스케일 특징을 제공하며, Focal Loss는 클래스 간 데이터 불균형 문제를 완화하는데 기여한다. 이를 통해 RetinaNet은 소수 클래스에서도 높은 정확도를 유지하면서 전체적인 탐지 성능을 향상시켰다.

3 Method

본 실험에서는 Single stage detector인 RetinaNet을 사용하였다. RetinaNet은 FPN과 Focal Loss를 활용하여 다양한 크기의 객체를 탐지하고 클래스 불균형 문제를 해결한다. 또한, 앵커 박스를 사용하여 중첩된 객체들을 효과적으로 탐지하며, NMS(Non-Maximum Suppression) 기법을 통해 중복된 Bounding Box를 제거하였다. NMS는 탐지된 객체 중 높은 신뢰도를 가진 박스를 선택하고, 그와 겹치는 낮은 신뢰도의 박스를 제거하는 방법으로, 중복된 객체를 제거하여 최종 출력의 정확성과 효율성을 높인다.

3.1 데이터셋

사용한 데이터셋은 자율주행을 위한 2D 객체 탐지 데이터셋인 KITTI이다. 데이터 전처리 과정에서 이미지를 비율을 유지하며 크기를 조정하고, 패딩을 추가하였다. 이러한 전처리 과정은 객체의 형태 왜곡을 방지하고, 네트워크가 다양한 크기의 입력 데이터를 효율적으로 처리할 수 있도록 돕는다. 패딩은 이미지 크기를 정규화하여 배치 처리 시 계산 효율성을 높이는 데 기여한다.

3.2 Backbone과 Loss

Backbone으로는 ResNet50을 사용하여 빠르고 효율적인 이미지 표현을 학습하였다. 또한, Classification Loss 계산 시에 Focal Loss를 적용하여 학습의 클래스 불균형 문제를 해결하고 Box Regression에는 Smooth L1 Loss를 적용하였다.

4 Result

RetinaNet 모델의 성능을 각 epoch 시점에서 비교하여 분석하였다. Loss와 Mean IoU를 기준으로 평가하였으며, 각 지표의 변화를 통해 모델의 학습 경향성을 확인하였다. Table 1은 5, 15, 30 epoch 시점의 Loss와 Mean IoU 값을 요약한 것이다.

epoch	loss	Mean IoU
5	1.0043	2.8445e-04
15	1.0451	2.8204e-04
30	1.2304	3.3542e-04

Table 1: Object Detection : Loss와 Mean IoU의 Epoch별 결과

Loss 기준으로 볼 때, 학습 초기에 빠르게 수렴하여 5 epoch 내에 최적화가 이루어진 것으로 나타났다. 반면, Mean IoU는 진동을 동반하며 점진적으로 상승하는 경향을 보였으나, 약 10 epoch 이후 과적합이 발생하는 양상을 관찰할 수 있었다. 이러한 결과는 모델이 초기에는 전반적인 손실을 줄이는 데 집중하였으나, 이후에는 세부적인 객체 탐지 성능 향상 과정에서 데이터 적합도가 과도해졌음을 시사한다.

정성적 분석을 위해 각 epoch에서의 탐지 결과를 시각적으로 비교하였다



Figure 1: Object Detection : Epoch별 Pedestrian 차이 (5 / 10 / 15 epoch)



Figure 2: : Object Detection : Epoch별 Van 차이 (5 / 10 / 15 epoch)

5 epoch에서는 보행자를 인식하지 못하는 사례가 다수 존재하였으나, 15 및 30 epoch에서는 이러한 객체가 성공적으로 탐지되는 결과를 보였다. 특히, 5 epoch와 15 epoch에서 제대로 탐지되지 않았던 밴 뒤의 컨테이너는 30 epoch에서 각각의 객체로 정확히 탐지되었다.

이러한 결과는 학습이 진행됨에 따라 FPN이 다양한 크기의 특성맵을 효과적으로 결합하면서 풍부한 정보를 수집한 데에 기인하는 것으로 판단된다. 이를 통해 모델은 보행자와 밴과 같은 다양한 객체를 구별할 수 있는 특징을 더 잘 학습하게 된 것으로 보인다. 정성적 평가를 바탕으로

Loss와 Mean IoU의 의미를 분석해 보면, Classification Loss와 Box Regression Loss가 합산되어있는 Loss는 학습 초기 단계에서 먼저 최적화가 이루어졌으며, IoU는 학습 후반부에 이르러 최적화가 완료된 것으로 해석할 수 있다.

5 Discussion

본 연구에서는 RetinaNet 모델을 활용하여 자율주행 보조장치를 구현하기 위한 실험을 수행하였다. 실험 결과, Loss와 Mean IoU에서 상반된 경향이 나타났으나, 이를 정성적 평가를 통해 분석함으로써 모델의 최적화 과정과 그 의미를 구체적으로 설명할 수 있었다. 이러한 분석은 모델의 성능을 깊게 이해하는 데에 기여했다.

향후 연구에서는 FPN과 Focal Loss 기법의 실질적인 효과를 보다 정량적으로 검증하거나, 다양한 데이터셋과의 비교를 통해 모델의 일반화 성능을 평가하는 방향으로 확장될 수 있다.

References

- [1] Lin, T. Y., Goyal, P., Girshick, R., He, K., Dollár, P. (2017). Focal Loss for Dense Object Detection. arXiv preprint arXiv:1708.02002.
- [2] Geiger, A., Lenz, P., Urtasun, R. (2012). Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In Conference on Computer Vision and Pattern Recognition (CVPR).