

깊은 신경망 기반 객체 인식기를 활용한 교통 약자 인식

조영완, 이명학, 김대희, 이재구*
국민대학교 소프트웨어학부

*jaekoo@kookmin.ac.kr

Recognizing Transportation Vulnerable Using Deep Neural Network Based Object Detection

Youngwan Jo, Myunghak Lee, Daehee Kim and Jaekoo Lee*
College of Computer Science, Kookmin University.

요 약

교통약자는 건널목을 건널 때 일반 사람들보다 더 긴 시간이 필요하므로 건널목에서 서 있는 교통약자를 인식하여 녹색등의 점등 시간을 늘려주는 것이 필요하다. 우리는 건널목에서 서 있는 교통 약자를 인식하기 위해 교통 약자 데이터 집합인 Mobility-aids[1]를 Faster RCNN[2] 모델과 YOLO v5[3] 모델에 학습시켜 평균 정밀도와 실행 시간을 비교하였다. 실험 결과로 평균 정밀도는 YOLO v5 모델이 Faster RCNN 모델보다 16.5% 더 좋은 평균 정밀도를 보였고 실행 시간에서도 각 영상당 0.011 초로 Faster RCNN 모델의 실행 시간 대비 93.7% 감소하였다. 건널목의 교통 약자를 실시간으로 인식하는 데 있어 YOLO v5의 빠른 객체 인식은 적합할 것으로 판단된다.

I. 서 론

현재 건널목에서 녹색 등의 점등 시간은 건널목의 길이에 건널목 진입 시간을 더하여 정해진다. 이는 일반 사람들에게는 충분한 시간이지만 교통 약자들에게는 비교적 짧게 느껴진다. 현재 건널목의 녹색등 점등 시간은 고정된 것이 아니므로 신호등마다 시간을 바꾸는 것이 가능하다. 우리는 객체 인식 기술을 이용하여 교통약자가 건널목에 들어설 때 녹색등의 점등 시간을 늘리는 방법을 제안한다. 이를 위해서는 객체 인식 모델의 정확한 교통약자 인식과 빠른 실행 시간이 필요하다.

본 논문에서는 이를 위해 객체 인식 모델들 간의 성능을 비교하였다. Faster RCNN[2] 모델과 YOLO v5[3] 모델을 Mobility-aids[1]의 교통약자 데이터 집합을 이용하여 학습하고 평균 정밀도(Average Precision)와 실행 시간(Inference Time)을 비교하였다.

5 개의 객체에 대한 평균 정밀도는 Faster RCNN 이

46.8%, YOLO v5는 54.5%로 YOLO v5 모델이 16.5% 더 좋은 성능을 나타냈다. 실행 시간은 YOLO v5 모델이 각 영상당 0.01초로 초당 100장의 객체를 인식할 수 있다는 것을 보여주었다. 이는 Faster RCNN과 비교하였을 때 실행 시간이 93.7% 감소하였다. 따라서 YOLO v5 모델이 교통약자 인식에 더 적합하다고 판단된다.

II. 본론

Faster RCNN은 2 단계 인식기(2-Stage Detector)로 RPN(Region Proposal Network)에서는 슬라이딩 윈도우(Sliding Window) 방식을 이용하여 후보 객체 영역(Region Proposal)을 약 2,000 개 생성하고 분류기(Classifier)에서는 RPN에서 추출한 영역에 대해 객체인지 아닌지 분류한다. 그러므로 많은 후보 객체 영역을 확인하기 때문에 평균 정밀도가 높고 겹쳐지거나

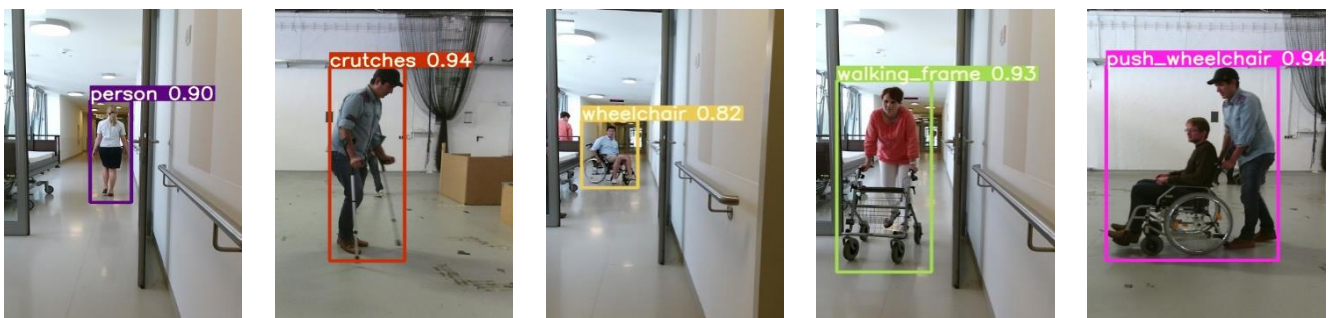


그림 1. 각 분류에 대한 YOLO v5[3]의 예측 결과

객체 인식기	각 부류에 대한 평균 정밀도(%)					모든 부류에 대한 평균 정밀도(%)	실행 시간 (s)
	사람	목발	휠체어	보행 보조기	휠체어 미는 사람		
Faster RCNN[2]	45.6	50.9	35.4	56.6	45.7	46.8	0.17
YOLO v5[3]	40.1	72.2	19.1	63.6	77.6	54.5	0.01

표 1. Faster RCNN[2]과 YOLO v5[3]의 Mobility-aid[1] 데이터 집합에 대한 테스트 성능 비교

작은 사물에 대해서도 인식률이 높다. 하지만 그만큼 실행 시간이 오래 걸린다.

YOLO v5는 단일 단계 인식기(1-Stage Detector)로 영상을 한번 보는 것으로 객체의 종류와 위치를 찾는다. 영상을 격자로 나누고 경계 상자를 예측하고 각 상자에 대한 신뢰도 점수를 예측한다. 영상 전체를 한 번에 처리하기 때문에 객체에 대한 맥락적 이해도가 높아서 배경에 대한 오류가 발생할 확률이 낮아지고 처리 과정이 단순하므로 매우 빠른 속도로 객체를 인식할 수 있다.

Mobility-aids의 데이터 집합은 사람, 목발, 휠체어, 보행보조기, 휠체어 미는 사람의 5개의 객체로 구성되어 있다. 이 데이터 집합은 스위스의 병원에서 로봇이 찍은 동영상 프레임 단위로 잘라 만들어졌다. 연속적인 동영상을 사용하여 만들어진 데이터 집합이기 때문에 새로운 장면이 등장하기 전까지는 비슷한 영상들로 구성되어 데이터의 양이 적은 것이 특징이다. 주석에는 각 영상에 대한 파일 이름, 경계 상자에 대한 각 꼭짓점의 위치, 각 객체의 거리 정보를 나타내는 깊이(Depth), 객체의 이름과 영상의 높이와 넓이가 쓰여 있다.

III. 실험

우리는 Mobility-aids의 훈련 집합 15,250 장, 테스트 집합 1,804 장을 훈련과 테스트 시에 사용했으며, 훈련 집합의 10%인 1,500 장을 추출하여 검증 집합으로 사용하였다.

Faster RCNN은 Detectron2[4] 패키지 중 COCO 데이터 집합으로 사전 학습(Pre-trained)된 Resnet 101[5] 모델을 중추신경망(Backbone Network)으로 사용했고 전이 학습(Transfer Learning)을 통해 훈련했다. YOLO는 v5 모델을 사용하였고 마찬가지로 COCO 데이터 집합으로 사전 학습된 매개변수를 기반으로 전이 학습을 통해 학습하였다. 실행 시간을 비교하기 위해 테스트 시 데이터 표본의 크기(Batch Size)는 1로 하였고 평균 정밀도를 비교하기 위해 IoU(Intersection over Union)의 임계값(Threshold)은 0.65로 하였다. 실행 환경은 NVIDIA TITAN XP를 사용하였다.

Faster RCNN과 YOLO v5의 평균 정밀도와 실행 시간은 [표 1]과 같다. YOLO v5가 Faster RCNN에 비해 평균 정밀도에서는 16.5% 더 좋은 성능을 보였고,

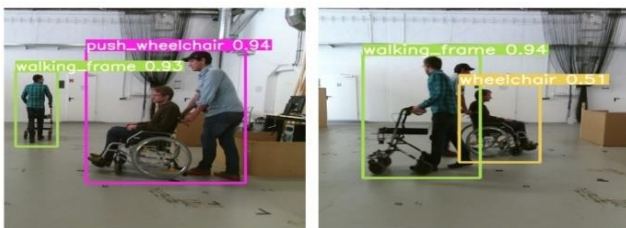


그림 2. 휠체어 미는 사람 객체를 휠체어 객체로 잘못 판단한 예시(오른쪽)

실행 시간은 93.7% 단축했다. [그림 1]은 YOLO v5 모델이 테스트 집합에 대해 예측한 결과이다.

데이터 집합의 주석에는 깊이에 관한 내용이 포함되어 있다. 하지만 우리는 객체의 깊이를 학습 시 사용하지 않았기 때문에 휠체어와 휠체어 미는 사람 객체는 서로 잘못 판단되는 경우가 많은 것을 확인하였다. [그림 2]는 YOLO v5 모델에서 휠체어 미는 사람 객체가 휠체어로 잘못 판단되는 경우를 보여주고 있다. 휠체어 객체가 휠체어 미는 사람 객체로 판단되는 경우가 많았기 때문에 두 모델 모두에서 휠체어에 대한 평균 정밀도가 낮은 것을 확인할 수 있다. 하지만 두 객체 모두 휠체어를 포함하는 경우이기 때문에 교통약자 인식에서는 문제가 되지 않는다고 판단된다.

교통약자를 발견하여 시간을 바꾸기 위해서는 평균 정밀도와 빠른 객체 인식 모두 중요하기 때문에 YOLO v5 모델이 Faster RCNN보다 더 적합하다고 판단된다.

IV. 결론

우리는 교통약자의 건널목 횡단을 돕기 위해 객체 인식 모델을 이용하여 교통약자를 찾아내기 위해 Faster RCNN과 YOLO v5의 평균 정밀도와 실행 시간을 비교하였다. YOLO v5가 Faster RCNN보다 16.5% 더 정밀하고 93.7% 더 빠른 것을 확인하였다. 따라서 건널목에서 객체 인식을 위한 모델로 YOLO v5가 더 적합하다고 판단된다.

ACKNOWLEDGMENT

이 성과는 2020년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원(No. NRF2018R1C1B5086441)과 과학기술정보통신부 및 정보통신기획 평가원의 SW 중심대학지원사업(2016-0-00021)으로 수행된 연구임.

참고 문헌

- [1] Andres Vasquez, Marina Kollmitz, Andreas Eitel and Wolfram Burgard, "Deep Detection of People and their Mobility Aids for a Hospital Robot", ECMR, 2017.
- [2] Shaoqing Ren, et al., "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", Microsoft Research, 2016.
- [3] Ultralytics, YOLO v5(2020), Retrieved June, 10, 2020, from <https://github.com/ultralytics/yolov5>
- [4] Facebook, Detectron2(2019), Retrieved April, 17, 2020, from <https://detectron2.readthedocs.io/index.html>
- [5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition", arXiv:1512.03385, 2015.
- [6] Lin, T.-Y., Dollar, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. "Feature pyramid networks for object detection", CVPR, 2017.