

시각장애인을 위한 시각 도움 서비스를 제공하는 인공지능 시스템
개발

오창현, 최광요, 이호영

한국 폴리텍 대학교 인천캠퍼스 메카트로닉스 학과

zansis23@naver.com/ckyh94111@naver.com /

hoyoung_Lee@kopo.ac.kr

Development of artificial intelligent system for
visual assistance to the Visually Handicapped

Changhyeon Oh, Gwangyo Choi ,Hoyoung Lee

Dept. of Mechatronics, Korea Polytechnics Incheon

요 약

현재 시각장애인들의 일상생활에 있어 많은 불편함을 겪고 있어 시각장애인에게 도움이 되고자 실시간 객체인식을 하여 보행환경의 정보를 전달하는 안경을 만드는 프로젝트를 진행하였다. 핵심 기능에 해당하는 객체인식은 인공지능 모델 YOLOv4가 사용되었으며, 시각장애인의 입장에서 걸어 다닐 때 인식 되어야 하는 객체들을 선정하고, 이들을 대상으로 학습 데이터를 재구성하고 YOLOv4의 재학습을 진행하였다.

학습 결과 모든 객체들에 대한 정확도는 68%를 보였으나 시각 장애인이 걸어다닐때 인식되어야 하는 필수객체(Person, Bus, Car, Traffic_light, Bicycle, Motorcycle)들의 인식률은 84%로 측정되었다. 향 후 진행될 학습에선 더욱 다양한 방법으로 학습데이터를 확보하고, YOLOv4가 아닌 darkflow를 이용해 다양한 parameter로 학습을 진행하여 다면적인 성능비교가 필요하다.

Currently, blind people are experiencing a lot of inconvenience in their daily lives. In order to provide helpful service for the visually impaired, this study was carried out to make a new smart glasses that transmit information monitoring walking environment in real-time object recognition. In terms of object recognition, YOLOv4 was used as the artificial intelligence model. The objects, that should be identified during walking of the visually impaired, were selected, and the learning data was populated from them and re-learning of YOLOv4 was performed.

As a result, the accuracy was average of 68% for all objects, but for essential objects (Person, Bus, Car, Traffic_light, Bicycle, Motorcycle) was measured to be 84%. In the future, it is necessary to secure the learning data in more various ways and conduct CNN learning with various parameters using darkflow rather than YOLOv4 to perform comparisons in the various ways.

1. 서론

최근 국내 마트나 음식점 방문시 안내견을 동반한 출입 제한 및 인도에 설치된 시각장애인을 위한 점자 블록을 철거하는 등 시각장애인의 일상생활이 점점 불편해지고 있다. 이러한 현재 상황에 시각장애인들에게 도움이 될 프로젝트를 구상하던 중 딥러닝을 이용하여 실시간으로 객체 인식을 하며 음성재생으로 시각장애인들에게 정보를 전달하는 방식의 해결책을 마련하였다. . 그 중 YOLO[1]를 본 프로젝트의 인공지능 모델로 선정하였다. YOLO는 한 장의 영상 사진안에 존재하는 여러 개의 객체들을 한번에 인식할 수 있는 기능을 제공하고 있다. 그리고 내부에 영상의 화질을 보완하기 위한 전처리 기능을 포함하고 있어서 영상의 잡음에도 강인한 성능을 보여주고 있다. 연산성능의 측면에서도 강력한 하드웨어를 활용하지 않더라도 준수한 성능으로 실시간 객체 인식을 가능하게 해준다. 무엇보다도, 딥러닝이라는 분야의 접근성 및 전문성으로 배우기 어렵다는 인식을 깨고 다양한 분야에서 활용이 되고 있으며 누구나 사용이 가능해 전문가가 아니어도 쉽게 weight를 만들고 사용 할 수 있다.

따라서 많은 사람들이 초기 설정값을 조절하고 다양한 데이터로 서로 다른 weight를 만들어 공유함으로써 weight의 성능과 질을 높이고 후에 시각장애인들의 일상생활에 도움이 되는 최종 weight가 산출됨으로서 시각장애인들의 불편함을 해소하는데 목표를 두었다.

2.재료 및 방법

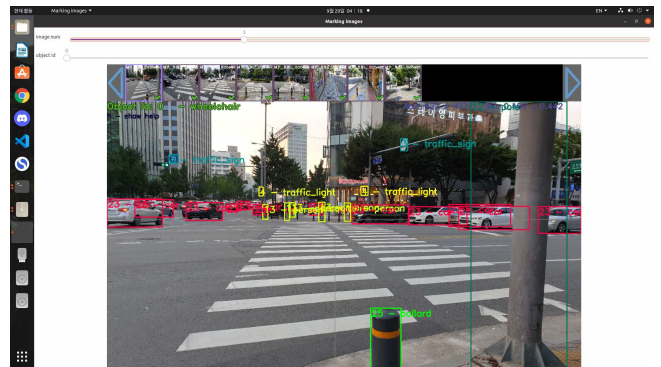
2.1. 학습 데이터

YOLOv4 모델의 학습에 사용된 데이터는 AI Hub[2]에서 제공하는 국내 도로 사진들의 사진과 Bounding Box 처리가 되어있는 xml파일에서 제공 받았다. 해당 데이터는 다양한 상황의 데이터를 인위적으로 생성하기 위해서 Image Augmentation를 이용해 학습 데이터를 추가하였으며, 그 결과 약 33만장의 사진과 47000장의 테스트 데이터로 구성되어 학습을 진행하였다. 인공지능이 감지해야 할 객체의 Class는 총 28개로, AI Hub에서 기본으로 설정된 Class로 학습을 진행 하였다. 주로 감지되는 객체는 시각장애인이 보행시 필요한 정보들인 Person, Bus, Car, Traffic_light, Bicycle, Motorcycle이다.

2.2. 이미지 전처리

첫 번째로, AI Hub로부터 제공된 데이터 중 학습에 투여될 데이터의 선정을 사진 속 신호등의 유무로 판단하였다. 그 이유로, 제공된 이미지 파일의 Bounding Box 정보 중 Traffic_light 인식이 횡단보도의 신호등과 교통 신호등 구분이 되어있지않아 선별 작업을 사전에 수행하였다.

본 프로젝트는 시각장애인의 입장에서 객체 인식을 진행하므로 먼저, 제공된 xml 파일의 label name 중 Traffic_light가 포함된 사진과 Bounding Box 좌표값을 각각 추출하여 작업에 사용될 데이터로 분류를 진행한 뒤 Traffic_light로 labeling이 되었던 Bounding Box 중 교통 신호등의 Bounding Box를 지워 횡단보도의 Traffic_light만 인식할 수 있도록 추가적인 라벨링 작업을 수행하고, 학습을 진행하였다. 또한 Ai Hub에서 제공한 xml파일의 Bounding Box 좌표 값 형식(CVAT형식[xtl, ytl, xbr, ybr])은 YOLO에서 요구하는 Bounding Box 좌표 값 형식[centerX, centerY, w, h]과 일치하지 않으므로 Yolo_mark[3]를 통해 기존의 Bounding Box의 좌표값을 YOLO와 호환되는 형식으로 변경하였다.

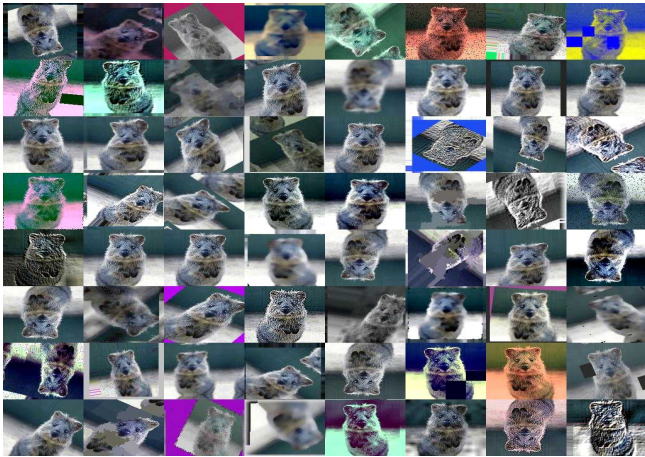


(그림1) Yolo mark를 이용한 교통신호와 신호등의 구분 및 Bounding Box 저장한다.

2.3. 데이터 확대(image Augmentation)

딥러닝을 진행할 때 직면하는 문제는 데이터의 양이다. 따라서 한정된 데이터를 변형 시켜 늘리는 방법이 요구되는데, image파일을 회전, 색 변형, 다른 데이터와 합성 등 다양한 방법으로 데이터를 늘린

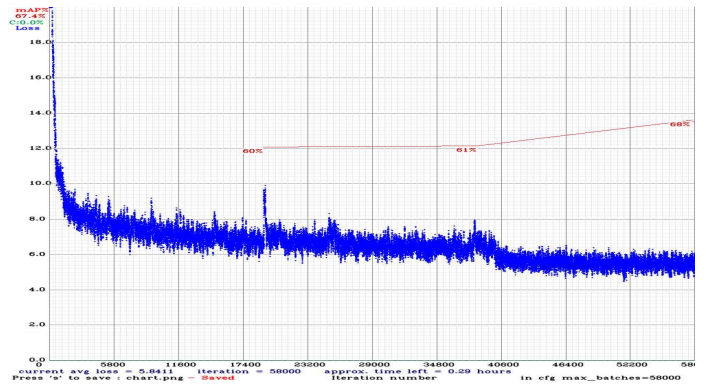
다. 본 연구에서 사용된 데이터 확대 방법으로 수직/수평으로 Flip, 이미지 크기의 -5%~10% crop, 색 반전, -45~+45도 각도로 회전, 이미지 Blur, 이미지의 밝기 조절을 하여 데이터 확대를 진행하였다. 그 결과 3만장의 사진을 추가하여 인공지능 학습에 사용하였다.



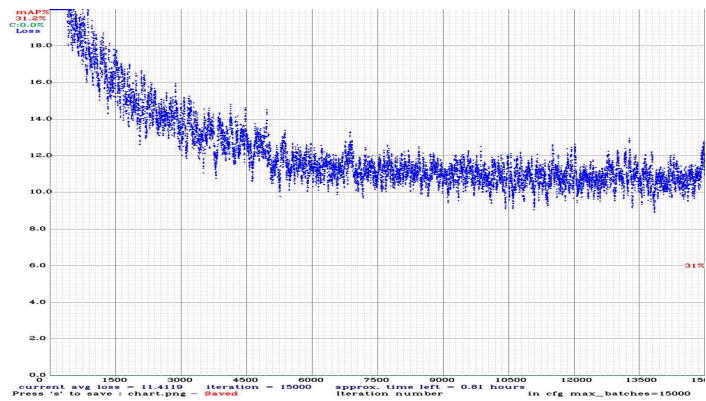
(그림 2) Image Augmentation 기법을 활용하여 한 장의 사진으로부터 여러개의 학습데이터를 확보하였다.

3. 결과 및 토론

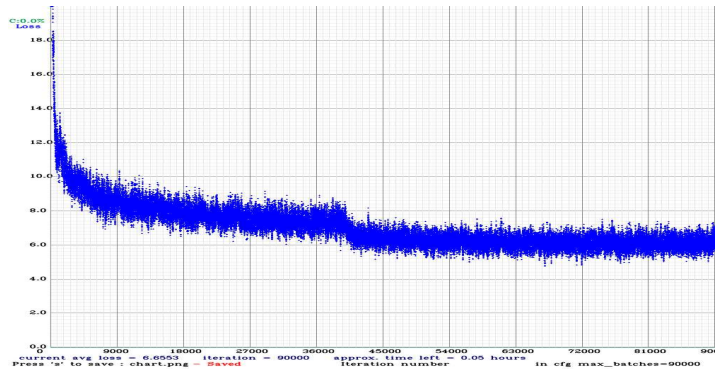
학습을 진행한 환경으로 GTX 970, RTX 3060ti, Amazone AWS(Tesla T4)였다. 각각 환경에 따라 특징이 뚜렷했는데, 하드웨어의 성능에 따라 weight의 성능과 작업시간이 많이 달랐다. 흥미로웠던 점은 AWS 와 Local(GTX 970, RTX 3060ti)에서 진행했던 3060Ti의 성능의 차이가 없었지만 시간적으로 Local환경이 학습을 진행하는데 소요되는 시간이 약 3배정도 빨랐다. 또한 SSD(Solid State Drive) 저장장치 환경을 제공하는 AWS에서의 학습과정과 달리 HDD(Hard Drive Disk)의 저장장치 환경에서 학습작업을 수행하였던 Local 환경하에서의 학습은 불안정한 결과를 보여주었다. GTX970를 이용한 학습과정은 학습 데이터를 읽어 들이고 저장하는 과정에서 병목현상이 발생하여 Training graph의 모습이 매우 불규칙하였다.



(그림 3) Amazon AWS상에서의 YOLOv4의 학습과정의 오류율 그래프



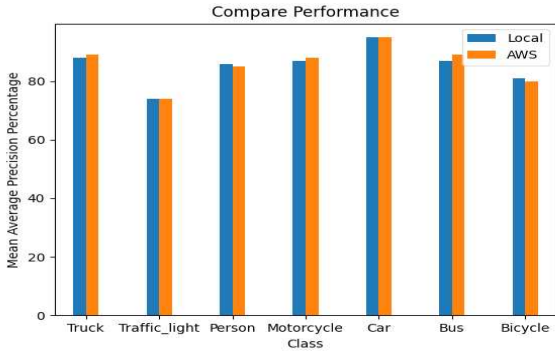
(그림 4) GTX970를 이용한 YOLOv4의 학습과정의 오류율 그래프



(그림 5) RTX 3060Ti를 이용한 YOLOv4의 학습과정의 오류율 그래프

amazon AWS, GTX970, 그리고 RTX 3060ti를 이용해 학습을 수행한 YOLOv4 모델의 평균 객체인식 성능은 각각 0.31, 0.68, 0.65로 산출이 되었으며, 이는 GPU성능과 저장소의 처리속도에 따라 성능차이를 보여주었다. 비록 전체 객체에 대한 평균 인식율은 60%대의 저조한 성능을 보였지만 시각장애인들의 입장에서 반드시 감지되어야 하는 객체들에 대해서는 인식율은 약 84%를 보여줌으로서 충분히 시각장애인의 안내에 활용될 수 있는 가능성을 보여주고

있다. 그래서 추후에 진행될 연구에서 Tensorflow와 darknet을 결합한 darkflow를 사용하여 기존 기본값으로 정해져있던 Parameter 값을 다양하게 조절하여 Training을 진행하며 성능 비교하여 성능을 좀 더 높이는 과정이 필요하다.



(그림3) AWS환경과 Local환경하에서 학습된 YOLOv4의 라벨별 인식 성능 비교

참고문헌

[1] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 6517-6525, doi: 10.1109/CVPR.2017.690.

[2] 한국지능정보사회진흥원, <https://aihub.or.kr/intro/policy>

[3] YOLO_mark, https://github.com/AlexeyAB/Yolo_mark

—본 논문은 과학기술정보통신부 정보통신창의인재 양성사업의 지원을 통해 수행한 ICT멘토링 프로젝트결과물입니다—