

에코빈 진행 보고서

목 차

작성 및 검토 확인란

구분	성명	년 월 일	서 명
작성자	이지훈	2023.05.23.	

개정 이력

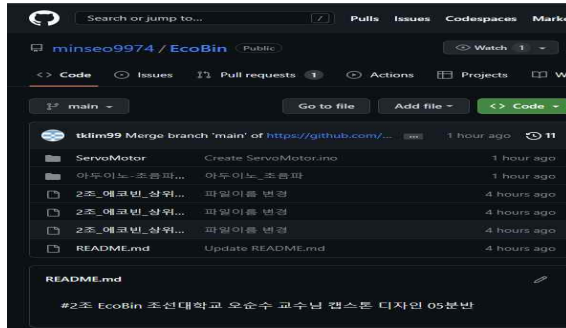
개정일자	버전	개정내용	작성자	확인자
2023.05.09	1.0	05/09 진도보고서	이지훈	-
2023.05.16	2.0	05/16 진도보고서	이지훈	-
2023.05.23	3.0	05/23 진도보고서	이지훈	-

제 1 장. 진도보고서	1
제 1 절 5월 9일 진도보고서	3
제 2 절 5월 16일 진도보고서	6
제 3 절 5월 23일 진도보고서	20

제 1 장. 진도보고서

제 1 절 5월 9일 진도보고서

- 버전관리와 협업을 위해 github 프로젝트를 생성함.
- 이후 <https://github.com/minseo9974/EcoBin>를 통해 진행 상황을 올리고 피드백 활동 진행함.



[그림 1-1] github 프로젝트 생성

[표 1] 에코빈 물품 조사

물품	가격 (원)	결제방식	수량 (개)
ESP32 IOT 사물인터넷 WIFI + 블루투스 듀얼모드 아두이노 보드 모듈	5,980	선불 결제 (네이버페이)	1
아두이노 ESP32 DevKitC V4 38p 블루투스 개발 보드	10,980(배송비 포함)		1
아두이노 SG-90 SG90서보모터	5,280		4
아두이노 스테핑 모터 키트	57,400(배송비 포함)		1
ESP32 WIFI + Bluetooth 일체형 개발보드	11,000	후불 결제 (에듀이노)	1
우노 R3 DIP 호환보드	23,000		2
초음파센서 HC-SR04	4,400		4
라즈베리파이 카메라모듈 V2	43,500		1
라즈베리파이 NOIR 적외선 카메라모듈	35,900		1

- 현재 를 제작을 위한 재료 제외, 기능 물품을 구매 완료 후 승인 대기 중 임.
- 선불 결제 : 79,640원, 후불 결제 : 117,800원, 총 197,440원을 사용함.

상태	작성일	결제방법	신청금액	작업
교수승인중	2023-05-02	현금영수증(사업자용)	79,640원	보기 취소
교수승인중	2023-04-28	전자(세금)계산서(청구/영수)	117,800원	보기 취소

[그림 1-2] 제작비 지급 신청 현황

- 에듀이노는 코딩 교육을 위한 교구, 전자부품을 전문적으로 취급하는 코딩 교구 전문 쇼핑몰임.
- 후불 결제 절차가 자세히 안내되어 있어 쉽게 이용할 수 있으며, 다른 학교에서도 많이 사용하는 쇼핑몰임.



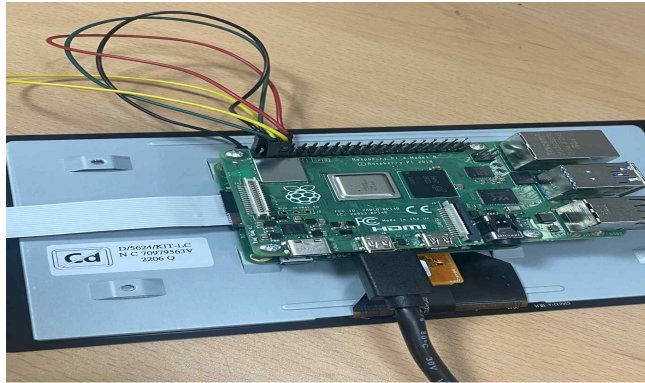
[그림 1-3] 에듀이노 후불결제

- 영상인식을 위한 라즈베리파이 OS를 <https://www.raspberrypi.com/software/>의 Raspberry Pi Imager v1.7.4를 이용하여 64비트 OS를 설치함.



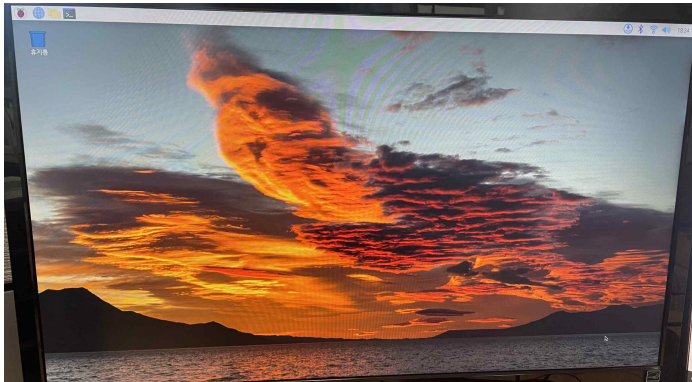
[그림 1-4] Raspberry Pi Imager

- 라즈베리파이 OS가 담긴 SD카드, 전원 연결하여 초기 라즈베리 환경을 설정함.



[그림 1-5] 라즈베리파이 연결

- 모니터와 라즈베리파이를 HDMI선으로 연결하여 모니터에서 라즈베리파이를 사용함.



[그림 1-6] 라즈베리파이와 연결된 모니터 화면

- 라즈베리파이를 사용 준비 완료함.

제 2 절 5월 16일 진도보고서

■ 지원금 신청 반려 재신청

· 반려사유 : 53,900 영수증이 없음. 또한 나머지 증빙이 잘못됨, 물품구매사진온 재료를 한곳으로 모은 후 1장으로 찍고, 영수증도 하나의 PDF로 묶어서 연번1번 칸에만 첨부. "연번1번"과 연번2번부터는 어떠한 증빙도 첨부하지 않겠.

[그림2-1] 반려사유

- 링크사업단에서 지원금 신청을 그림[2-1]와 같은 사유로 반려함.
- 서류를 재수정하는 과정 중 본인의 실수로 신청취소가 진행되었음.
- 현재 재 신청 완료 후 교수님 승인 대기 상태임.

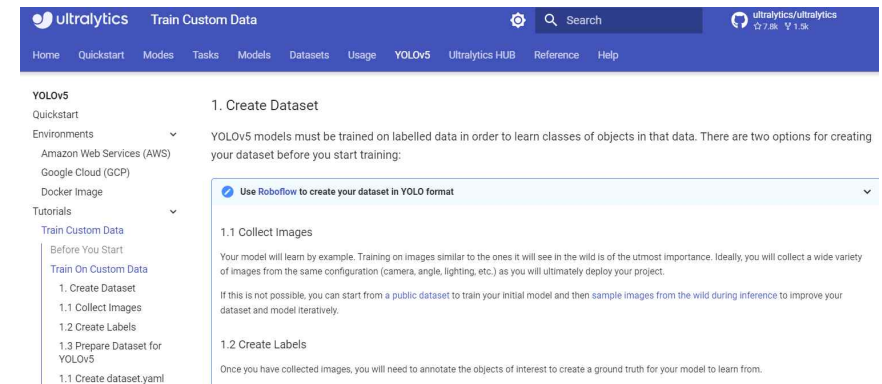
■ 데이터 세트 수집 및 구축

- 데이터세트 수집은 2가지 방법으로 진행함

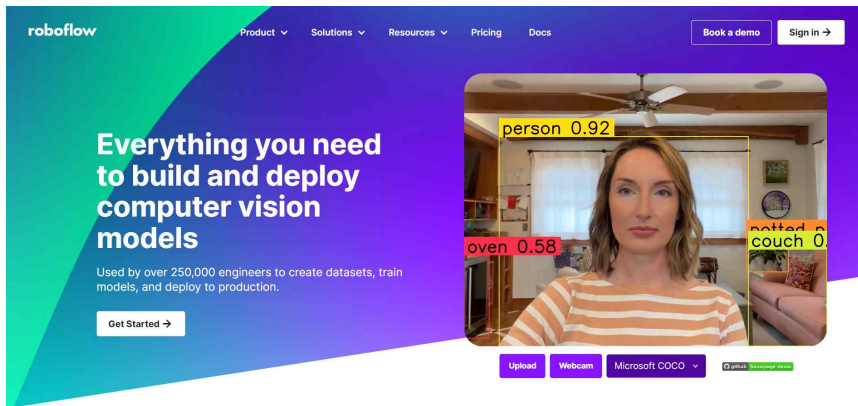
- YoloV5의 github에서 추천하는 roboflow 홈페이지에서 Yolo형식의 데이터세트 다운로드
- Ai허브에서 신청한 '생활 폐기물 이미지' 중 플라스틱과 캔 이미지 다운로드, labeling 프로그램을 통한 이미지 라벨링 작업

Roboflow (<https://roboflow.com/>)

- roboflow는 YOLO Github(<https://github.com/ultralytics/yolov5/>)에서 Custom Data를 만들기 위해 추천하는 사이트임.
- 다양한 컴퓨터 비전 모델을 만들기 위해 원시 이미지를 사용자 지정 형식에 따라 제공함.



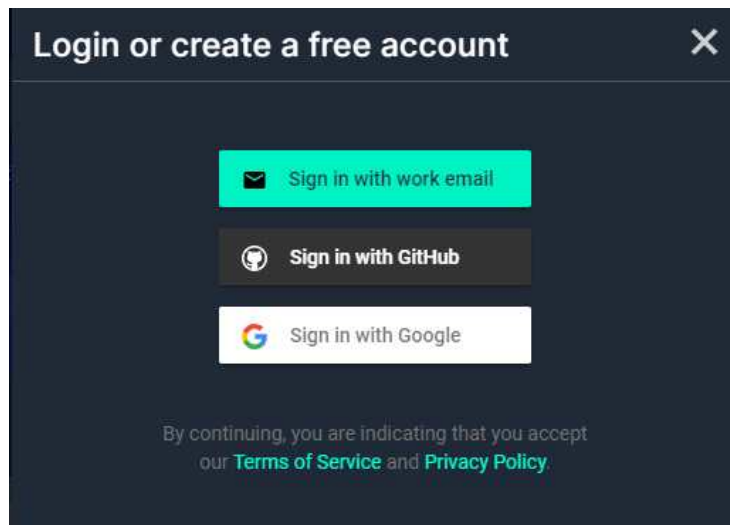
[그림2-2] Yolo에서 추천하는 Roboflow



[그림 2-3] Roboflow 홈페이지

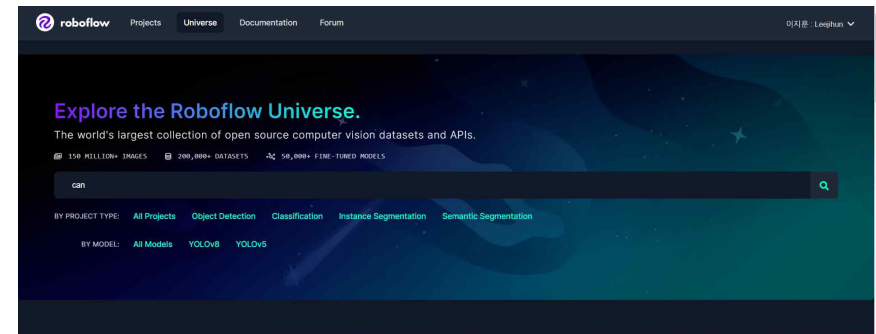
■ roboflow 데이터 수집 과정

- roboflow 회원 가입



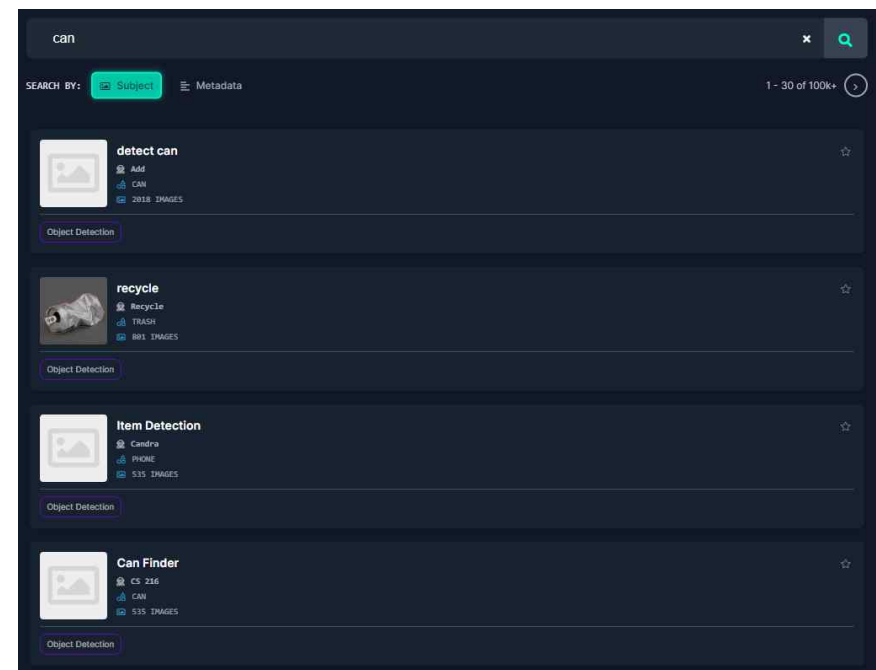
[그림 2-4] roboflow 회원가입

- Roboflow의 Universe 접속함
- 검색창에 찾고자 하는 데이터 검색함 - can검색



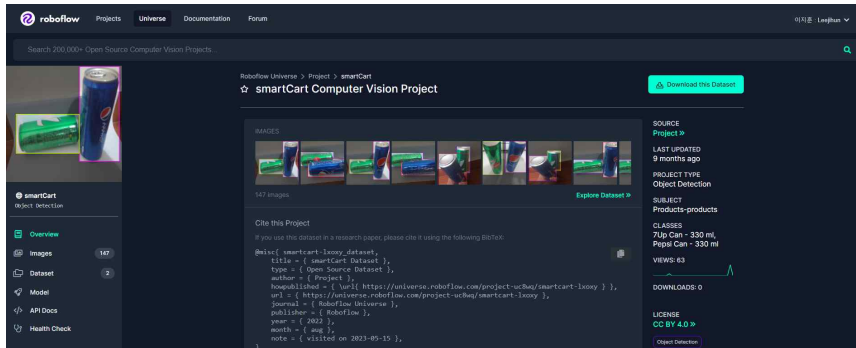
[그림 2-5] Roboflow Universe 검색

- 다른 사용자가 공유한 데이터셋 및 이미지가 나열됨



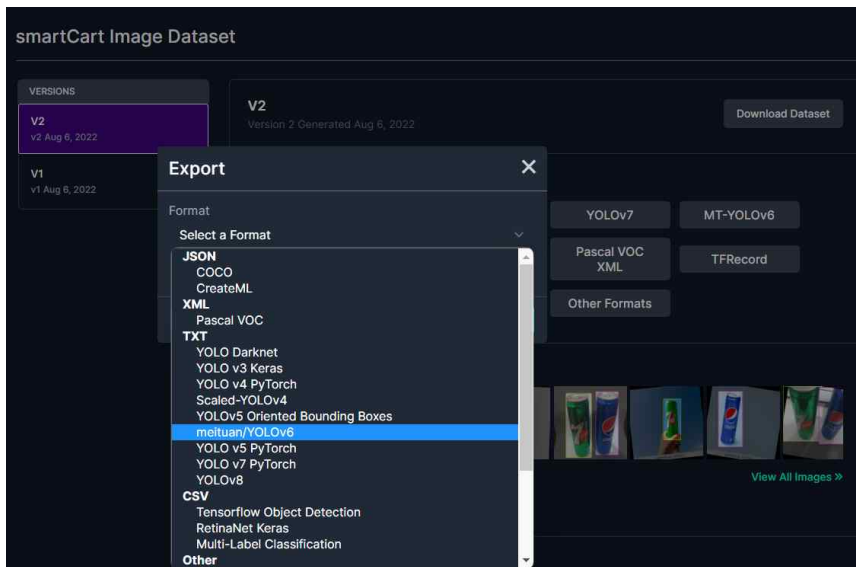
[그림 2-6] can 검색 결과

- 원하고자 하는 데이터셋, 이미지를 선택하여 다운로드 진행



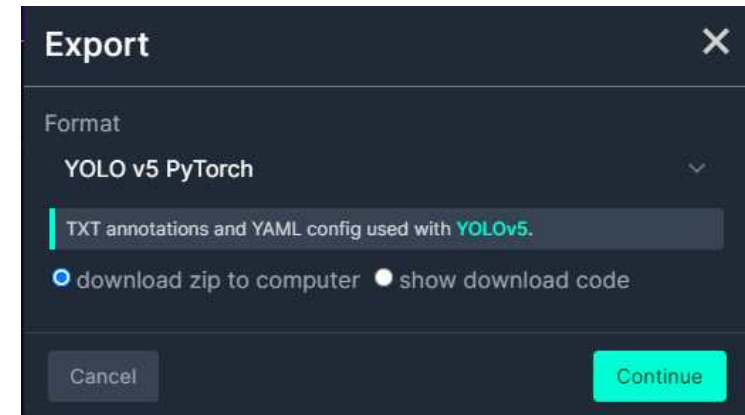
[그림 2-7] 선택한 can 데이터셋

- Download this Dataset을 클릭하면 사용자가 사용하려는 모델의 데이터형식을 설정할 수 있음.



[그림 2-8] 데이터 형식 설정

- download zip to computer : 컴퓨터에 zip형식으로 다운로드
- show download code : code를 실행해 다운로드

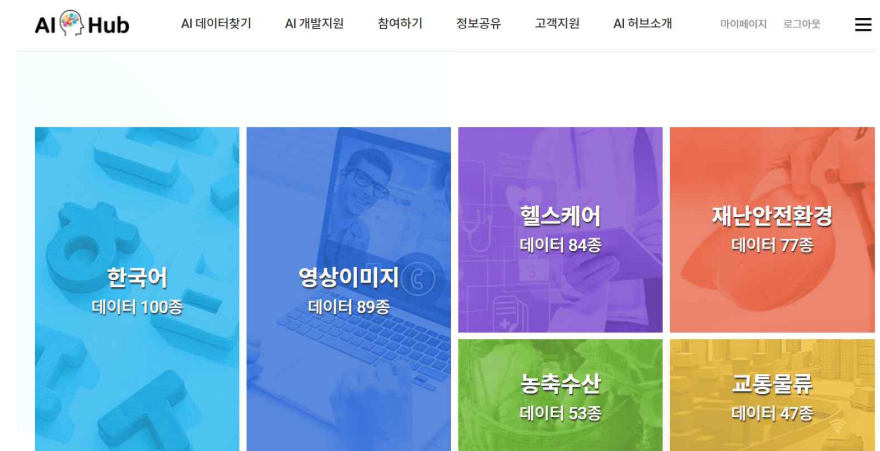


[그림 2-9] 다운로드 방식

- 컴퓨터에 zip형식으로 데이터셋을 다운 받아 수집 진행함.

■ Ai hub(<https://aihub.or.kr/>)

- AI hub는 기기기술 및 제품·서비스 개발에 필요한 AI 인프라(AI데이터, AI SW API, 컴퓨팅자원)을 지원하는 AI 통합 플랫폼
- 한국어, 영상이미지, 헬스케어, 재난안전환경 등 AI학습용 다양한 데이터를 제공함.



[그림 2-10] Ai hub

■ Ai허브 데이터 수집 과정

- 사용하고 싶은 데이터를 제공하는 절차에 맞춰 신청함.

데이터명	생활폐기물 데이터 활용-원류
유형	이미지
신청목적	<input type="text" value="선택"/>

☐ 본인은 AI 허브 데이터를 이용함에 있어 다음과 같은 사항을 준수할 것에 동의합니다. (필수)

1. 본 시데이터 등을 이용할 때에는 반드시 한국지능정보사회진흥원의 사업결과임을 밝혀야 하며, 본 시데이터 등을 이용한 2차적 저작물에도 동일하게 밝혀야 합니다.
2. 국외에 소재하는 법인, 단체 또는 개인이 시데이터 등을 이용하기 위해서는 수행기관 등 및 한국지능정보사회진흥원과 별도로 합의가 필요합니다.
3. 본 시데이터 등의 국외 반출을 위해서는 수행기관 등 및 한국지능정보사회진흥원과 별도로 합의가 필요합니다.
4. 본 시데이터는 인공지능 학습모델의 학습용으로만 사용할 수 있습니다. 한국지능정보사회진흥원은 시데이터 등의 이용의 목적이나 방법, 내용 등이 위법하거나 부적합하다고 판단될 경우

※ 데이터 다운로드 PC에서만 가능합니다.

다운로드

[그림 2-11] 데이터 신청

- 데이터 신청 완료하여 다운로드를 받을 수있음

데이터 신청 내역

데이터 다운로드 신청 절차		1.접수완료 : 데이터 신청 건이 접수된 상태입니다. 3.자동승인 : 데이터 다운로드가 자동승인 된 상태입니다. ※ 안심존 데이터의 경우 안심존 사용신청 상태가 제공됩니다	2.승인완료 : 데이터 다운로드 승인이 완료된 상태입니다. 4.반려 : 데이터 다운로드 신청이 반려된 상태입니다.
-----------------------	--	---	--

번호	분야	데이터명	구분	파일유형	상태	다운로드
1	재난안전환경	생활 폐기물 이미지		비디오, 이미지	자동승인	다운로드

[그림 2-12] 데이터 승인

- 데이터는 학습용과 검증용 데이터로 구분되며 학습용 데이터를 수집함.

<input type="checkbox"/>	NAME ▲	SIZE ▲
<input type="checkbox"/>	생활 폐기물 이미지	
<input type="checkbox"/>	Training	
<input type="checkbox"/>	Validation	

선택 다운로드

전체 다운로드

[그림 2-13] 다운로드 내용

- 다운로드를 진행함.

<input type="checkbox"/>	NAME ▲	SIZE ▲
<input type="checkbox"/>	[원천]의류_원피스_원피스.zip	651MB
<input type="checkbox"/>	[원천]나무_포장재_포장재.zip	651MB
<input checked="" type="checkbox"/>	[원천]플라스틱류_기타_기타.zip	673MB
<input type="checkbox"/>	[원천]전자제품_오디오_오디오.zip	674MB
<input type="checkbox"/>	[원천]형광등_환형_환형.zip	690MB
<input type="checkbox"/>	[원천]가구류_소파_소파.zip	692MB
<input type="checkbox"/>	[원천]캔류_스팸류_스팸류.zip	701MB
<input type="checkbox"/>	[원천]도끼류_도끼류_도끼류.zip	732MB

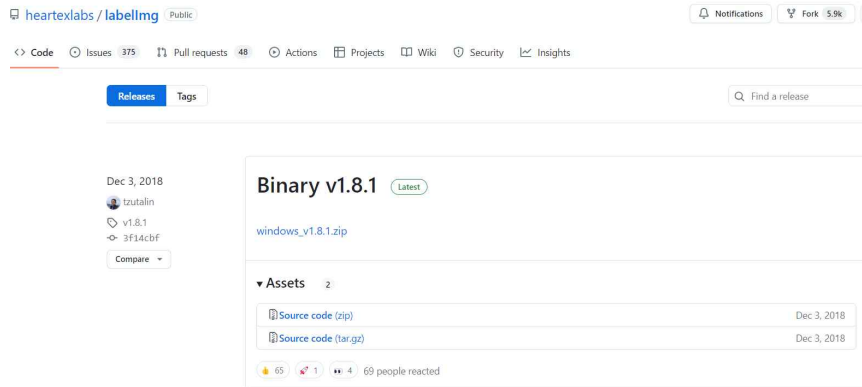
선택 다운로드

전체 다운로드

[그림 2-14] 데이터 다운로드

■ Ai hub 이미지 라벨링 작업

- 원시 이미지를 Yolo에 적용하기 위해 이미지 라벨링 작업이 필요함.
- <https://github.com/heartexlabs/labelimg/releases>에서 제공하는 labeling 프로그램을 이용함.



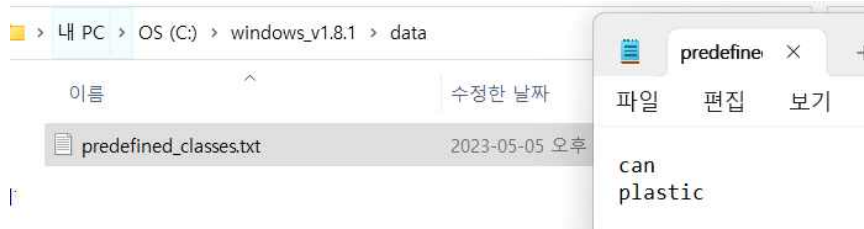
[그림 2-15] labeling 프로그램 git hub

- Binary v1.8.1 zip 파일을 다운로드함.



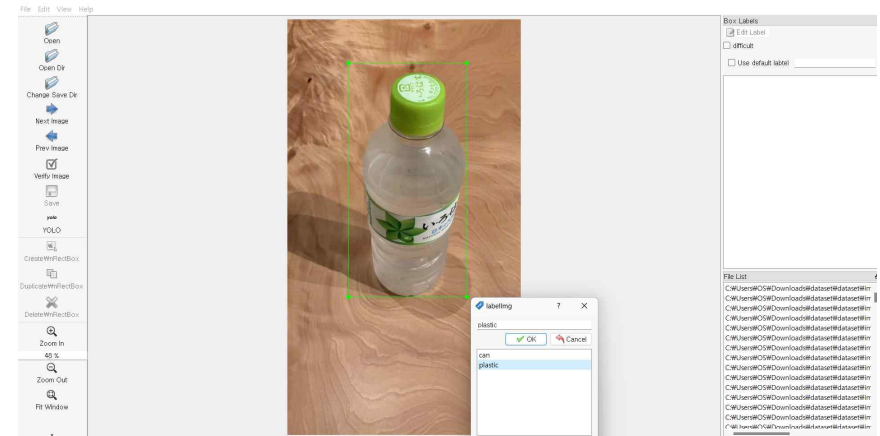
[그림 2-16] labeling 파일

- 라벨링 작업의 클래스를 구별 하기위해 data -> predefined_classes.txt 파일의 내용을 수정함.
- 캔, 플라스틱 두 개의 클래스로 라벨링 작업을 실시함.



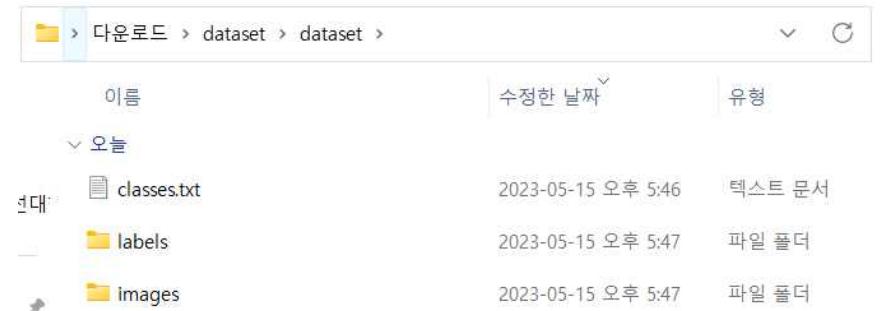
[그림 2-17] labeling 클래스 수정

- labeling 프로그램을 실행 후, 'YOLO'데이터 파일 형식으로 변경 뒤 라벨링 작업을 실시함
- plastic과 can의 클래스를 선택 후 폴더에 저장하여 image파일과 label파일을 분리함.

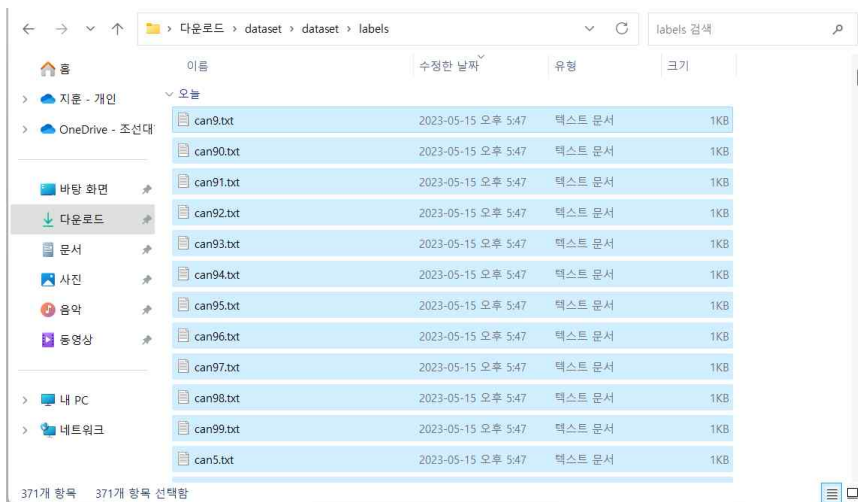


[그림 2-18] 라벨링 작업

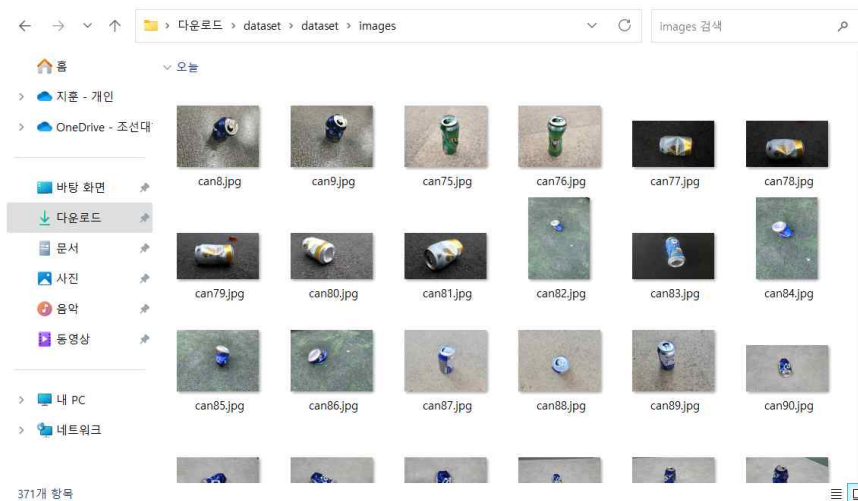
- classes.txt 파일은 사용된 클래스를 설명하는 텍스트 문서임
- labels는 labelimg파일을 통해 도출된 label의 txt문서를 저장함
- image는 라벨링하기 위한 원시 이미지 데이터임



[그림 2-19] 데이터세트 파일



[그림 2-20] 라벨링 txt데이터

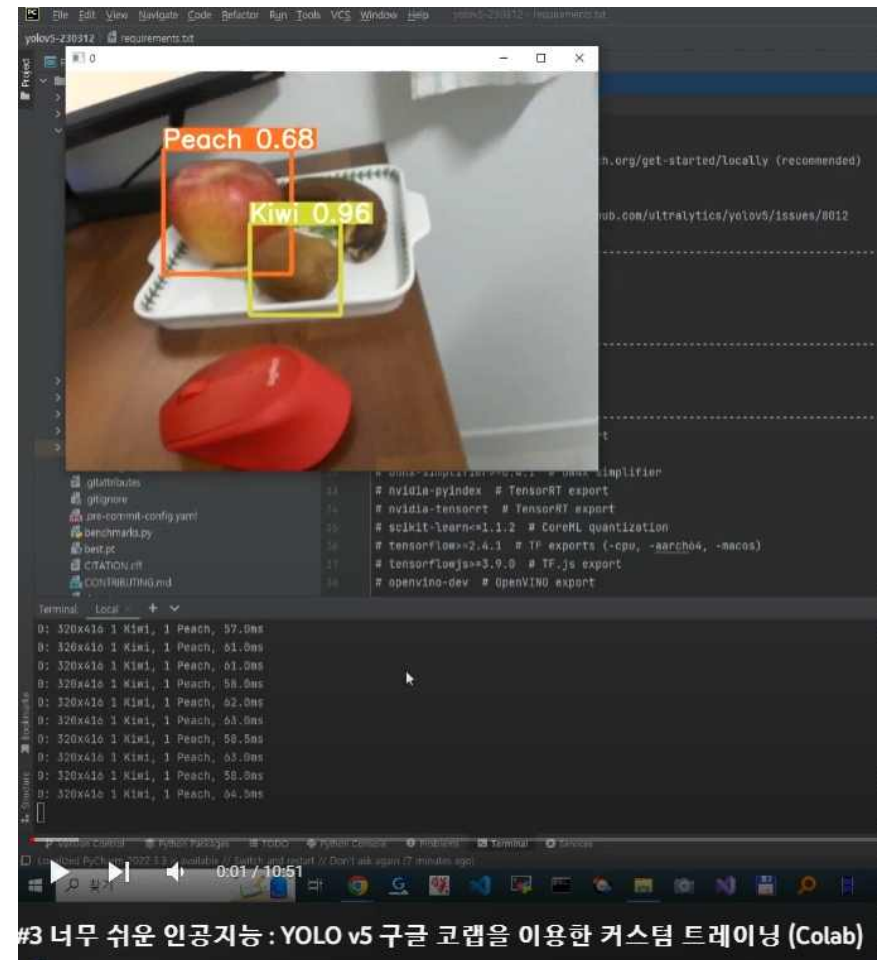


[그림 2-21] 캔과 플라스틱 이미지 데이터

- 현재 캔 200개, 플라스틱 171개를 수집하여 완료하였음.
- 각 팀원들이 700개씩 수집하여 총 2100개(캔 1050개, 플라스틱 1050개)의 데이터를 YOLO 모델에 입력하여 1차적으로 분류를 진행할 예정임.

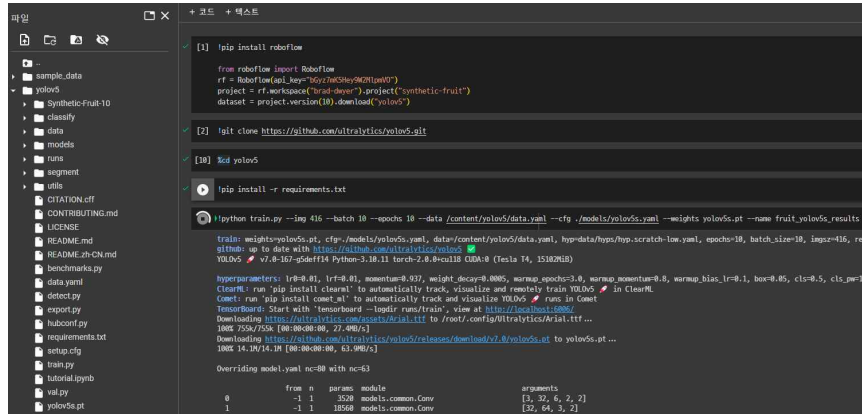
■ Yolo v5 시연

- 데이터셋 구축 후 <https://www.youtube.com/watch?v=U-wkRQ8U3GE>에서 YOLO v5의 영상을 보며 코랩으로 Yolo v5 모델 사용법을 익힘.



[그림 2-22] 구글 코랩을 이용한 커스텀 트레이닝

- roboflow의 synthetic-fruit 데이터셋을 이용하여 과일을 분류하는 yolo모델을 시연



[그림 2-23] yolo모델 시연

- 학습 진행 상황을 확인할 수 있음

Epoch	GPU_mem	box_loss	obj_loss	cls_loss	Instances	Size
0/9	1.14G	0.06566	0.0385	0.1004	40	416: 100% 1000/1000 [03:40<00:00, 4.54it/s]
		Class Images	Instances	P R	mAP50	mAP50-95: 100% 50/50 [00:12<00:00, 4.09it/s]
		all 1000	2803	0.00804 0.755	0.0155	0.00444
Epoch 1/9	1.14G	0.04707	0.03492	0.09724	45	416: 100% 1000/1000 [03:26<00:00, 4.85it/s]
		Class Images	Instances	P R	mAP50	mAP50-95: 100% 50/50 [00:10<00:00, 4.79it/s]
		all 1000	2803	0.0068 0.55	0.0104	0.0024
Epoch 2/9	1.14G	0.0442	0.03326	0.09645	62	416: 100% 1000/1000 [03:21<00:00, 4.96it/s]
		Class Images	Instances	P R	mAP50	mAP50-95: 100% 50/50 [00:09<00:00, 5.08it/s]
		all 1000	2803	0.0112 0.853	0.0267	0.0087
Epoch 3/9	1.14G	0.0411	0.0321	0.09493	43	416: 100% 1000/1000 [03:23<00:00, 4.92it/s]
		Class Images	Instances	P R	mAP50	mAP50-95: 100% 50/50 [00:10<00:00, 4.69it/s]
		all 1000	2803	0.0115 0.891	0.0482	0.0163
Epoch 4/9	1.14G	0.04014	0.03154	0.09146	25	416: 100% 1000/1000 [03:22<00:00, 4.95it/s]
		Class Images	Instances	P R	mAP50	mAP50-95: 100% 50/50 [00:09<00:00, 5.04it/s]
		all 1000	2803	0.481 0.204	0.105	0.0425
Epoch 5/9	1.14G	0.03965	0.03112	0.08618	58	416: 100% 1000/1000 [03:20<00:00, 4.99it/s]
		Class Images	Instances	P R	mAP50	mAP50-95: 100% 50/50 [00:10<00:00, 4.70it/s]
		all 1000	2803	0.368 0.353	0.159	0.0643
Epoch 6/9	1.14G	0.03879	0.03055	0.08197	37	416: 100% 1000/1000 [03:25<00:00, 4.86it/s]
		Class Images	Instances	P R	mAP50	mAP50-95: 100% 50/50 [00:10<00:00, 4.89it/s]
		all 1000	2803	0.336 0.42	0.212	0.0892
Epoch 7/9	1.14G	0.03791	0.03007	0.07886	51	416: 100% 1000/1000 [03:26<00:00, 4.85it/s]
		Class Images	Instances	P R	mAP50	mAP50-95: 100% 50/50 [00:10<00:00, 4.92it/s]
		all 1000	2803	0.375 0.44	0.282	0.108

[그림 2-24] 학습 진행과정

- 분류 결과 사과 이미지를 Redcurrant로 0.21확률을 인식하는 것을 확인함
- 데이터베이스를 살펴본 결과 과일 이미지 뿐만 아니라 과일이 아닌 이미지가 많아 질 좋은 데이터로 학습되지 않음을 확인함
- 질 좋은 데이터를 수집하여 YOLO v5 시연 경험을 통해 앞으로 캔과 플라스틱을 분류를 수행할 예정임.

(라즈베리웬캠 연결, 웹캠 출력 결과

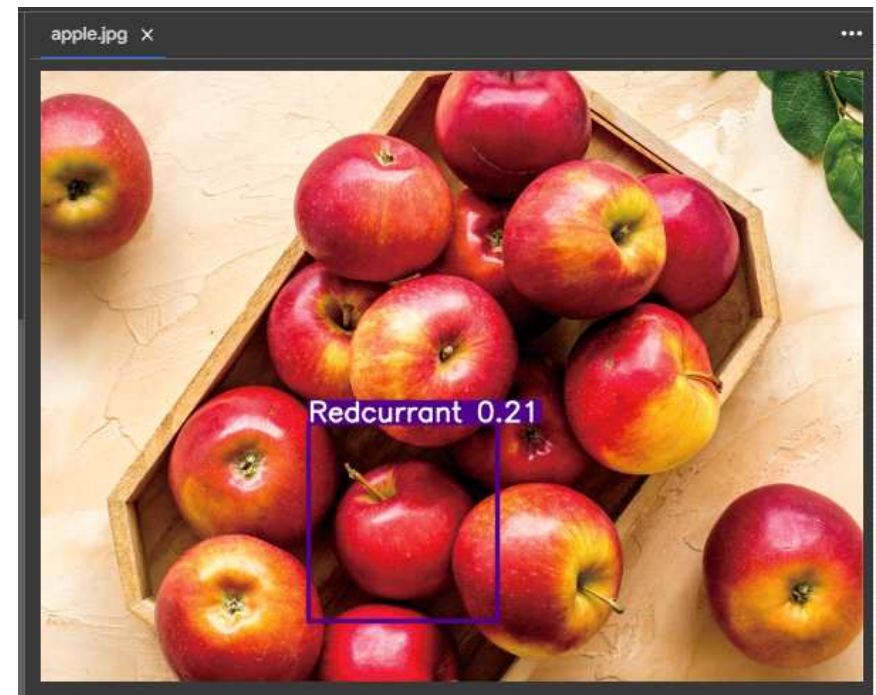
훈련 700, 검증 250개 때의 분류 결과

데이터수집완료 훈련 2100, 검증 400개 결과

YOLO성능 그래프(혼동행렬, FR)

제작비 환불결제 서류 완료

를 제작 표



[그림 2-25] YOLO V5 시연 결과

제 3 절 5월 23일 진도보고서

■ 제작비 정상 승인

2주차에 재신청한 전자(세금)계산서(청구/영수) - 후불결제 건이 정상 승인 완료함.

3주차를 진행하는 중 현금영수증(사업자용) - 선불결제 건이 정상 승인 완료함.

총 재료비 200,000 중 2,560원 남음.

● 재료비 금액 : 200,000 원 / 신청가능금액 : 2,560 원
 * 교수승인금 상태는 교수가 승인을 해야되는 상태입니다.
 * 승인대기 상태는 최종관리자가 승인을 해야되는 상태입니다.
 * [취소]는 최종관리자가 승인전까지 가능합니다.
 * 반려가 될 경우 [보기] 버튼을 누른 후 상세내역에서 재신청이 가능합니다.

상태	작성일	결제방법	신청금액	작업
재신청	2023-05-15	현금영수증(사업자용)	79,640원	보기 프린트
재신청	2023-04-28	전자(세금)계산서(청구/영수)	117,800원	보기 프린트

[그림 3-1] 제작비 정상 승인

■ 쓰레기통 예상 규격

현재 쓰레기통의 상,하,우 중 한 면은 투명하게 제작하여 쓰레기통이 분류되는 과정을 시각적으로 확인할 예정임.

[표 3-1] 쓰레기통 예상 규격 표

부품 \ cm	가로	세로	높이(두께)
외부 쓰레기통	40	40	60
내부 쓰레기통(2개)	16	40	40
상판 개폐 롤	18	40	3
물체 이동판	3	18	5
상판 중심부	8	40	3

■ 데이터 세트 구축 완료

캔과 플라스틱의 데이터 세트를 구축 완료함.

AI허브와, roboflow를 통해 수집하여 labeling프로그램을 통해 직접 라벨링 작업을 실시함.

[표 3-2] 데이터 세트 구축

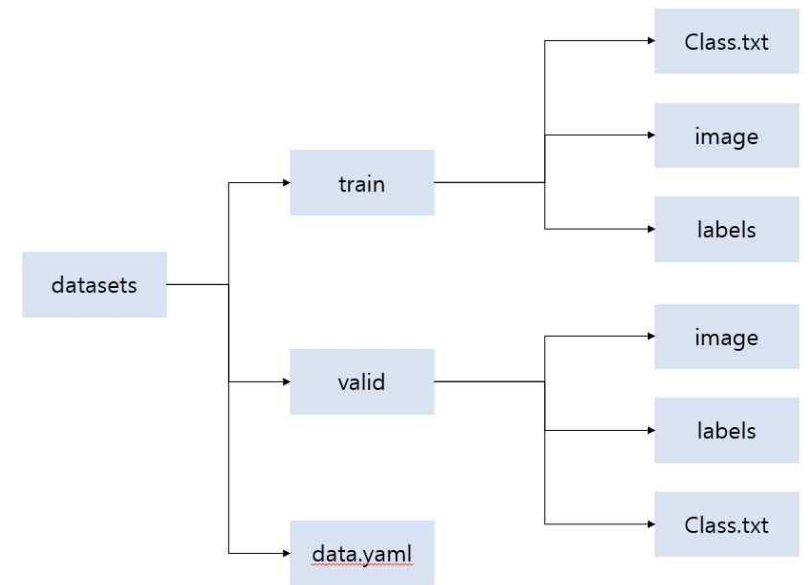
수집자	훈련데이터		검증데이터		합계
	캔(개)	플라스틱(개)	캔(개)	플라스틱(개)	
이민서	350	350	50	50	800
이지훈	350	350	100	100	900
임태경	350	350	50	50	800
합계	1050	1050	200	200	2500

데이터세트폴더는 train(훈련), valid(검증)데이터 폴더와 YOLO모델이 데이터를 불러올 수 있게 참고하는 data.yaml 파일이 있음.

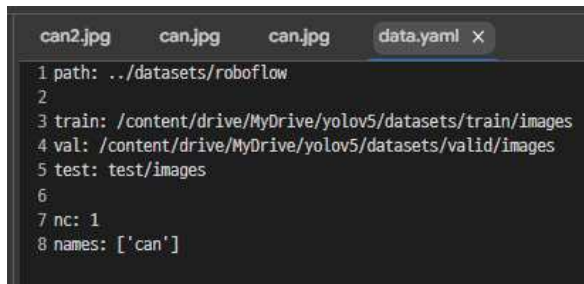
data.yaml 파일은 train, valid의 경로가 저장됨.

Class.txt 파일은 데이터의 클래스 이름이 저장되어있음.

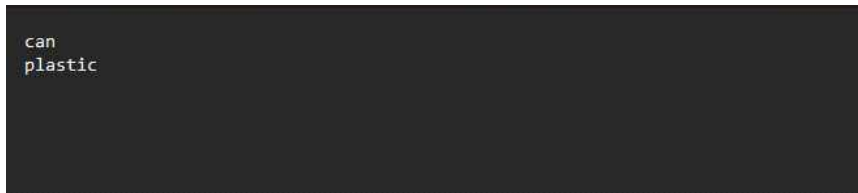
image와 labels 폴더는 각 훈련, 검증의 이미지와 라벨링된 txt파일로 구성됨.



[그림 3-2] 데이터세트 파일 구조



[그림 3-3] data.yaml 파일



[그림 3-4] class.txt

- 라즈베리파이 카메라 모듈V2 연결 및 웹캠 점검
라즈베리파이로 카메라 모듈V2를 연결하였음.

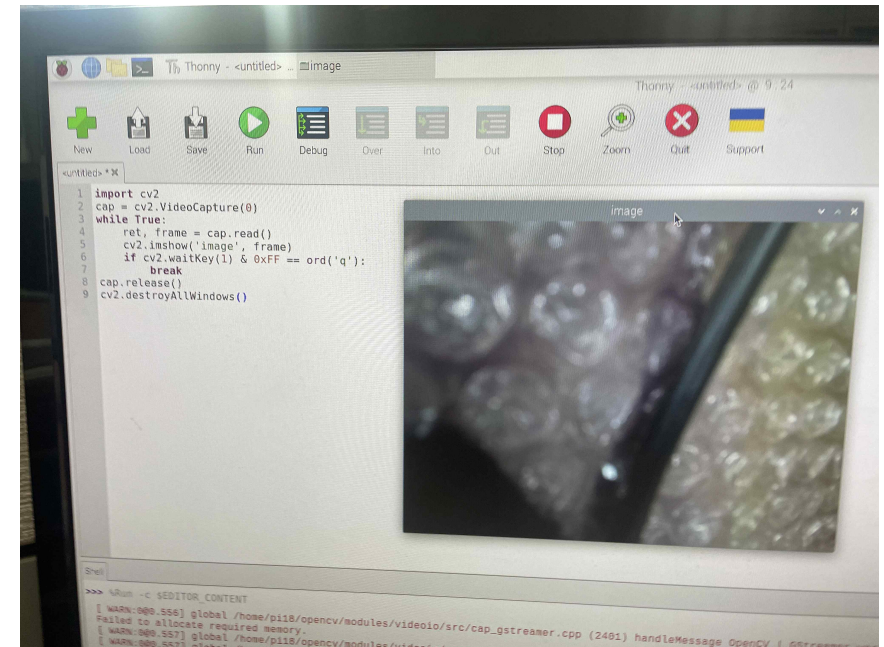


[그림 3-5] 라즈베리파이 카메라 모듈V2 연결

OpenCV4.5.5 버전을 라즈베리파이에 다운 받음.

1시간~1시간 30분의 경과시간이 소요됨.

이후 Throny 코딩환경에서 OpenCV를 이용해 카메라모듈을 실행함



[그림 3-6] OpenCV를 이용한 카메라 출력

카메라 작동 상태에 이상이 없었고 차후 캔과 플라스틱을 분류하기 위해 쓰일 예정임.

■ YoloV5 모델 구현 결과

처음 구현하였을때는 본인이 임의로 구축한 검증데이터 50개와 캔(100개), 플라스틱(100개) 총 250개의 훈련데이터로 모델을 학습하였음.

코딩환경은 코랩을 이용하여 구글 드라이브를 연동하였고, 구글 드라이브에 YoloV5 파일과 데이터셋을 저장하였음.

이를 통해 코랩환경에서 구글 계정을 연동시키면 코드 구현이 가능함.

이름	소유자	마지막으로 수정한 날짜	파일 크기
car1000.jpg	나	오늘 4:38 나	262KB
car1001.jpg	나	오늘 4:38 나	327KB
car1002.jpg	나	오늘 4:38 나	263KB
car1003.jpg	나	오늘 4:38 나	293KB
car1004.jpg	나	오늘 4:38 나	284KB
car1005.jpg	나	오늘 4:38 나	193KB
car1006.jpg	나	오늘 4:38 나	147KB
car1007.jpg	나	오늘 4:38 나	210KB
car1008.jpg	나	오늘 4:38 나	220KB
car1009.jpg	나	오늘 4:38 나	225KB
car1010.jpg	나	오늘 4:38 나	234KB
car1011.jpg	나	오늘 4:38 나	193KB
car1012.jpg	나	오늘 4:38 나	229KB

[그림 3-7] 데이터셋 드라이브 저장

200개의 훈련데이터와 50개의 검증데이터를 이용해 YoloV5 모델을 학습 시켰음.

사용된 라이브러리

```
# YOLOv5 requirements
# Usage: pip install -r requirements.txt
```

```
# ----- Base
```

```
gitpython>=3.1.30
matplotlib>=3.3
numpy>=1.18.5
opencv-python>=4.1.1
Pillow>=7.1.2
psutil # system resources
PyYAML>=5.3.1
requests>=2.23.0
scipy>=1.4.1
thop>=0.1.1 # FLOPs computation
torch>=1.7.0 # see https://pytorch.org/get-started/locally (recommended)
torchvision>=0.8.1
tqdm>=4.64.0
ultralytics>=8.0.100
# protobuf<=3.20.1 # https://github.com/ultralytics/yolov5/issues/8012
```

```
# ----- Logging
```

```
# tensorboard>=2.4.1
# clearml>=1.2.0
# comet
```

```
# ----- Plotting
```

```
pandas>=1.1.4
seaborn>=0.11.0
```

```
# ----- Export
```

```
# coremltools>=6.0 # CoreML export
# onnx>=1.10.0 # ONNX export
# onnx-simplifier>=0.4.1 # ONNX simplifier
# nvidia-pyindex # TensorRT export
# nvidia-tensorrt # TensorRT export
# scikit-learn<=1.1.2 # CoreML quantization
# tensorflow>=2.4.0 # TF exports (-cpu, -aarch64, -macos)
# tensorflowjs>=3.9.0 # TF.js export
# openvino-dev # OpenVINO export
```

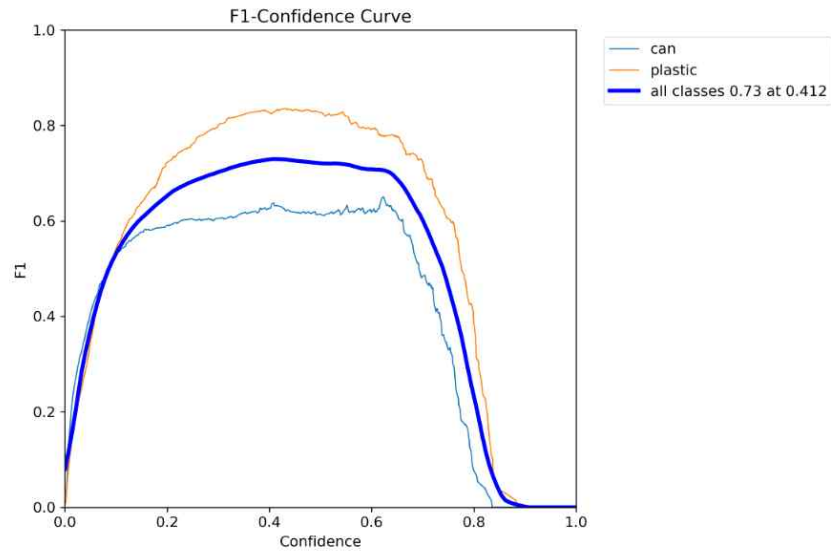
```
# ----- Deploy
```

```
setuptools>=65.5.1 # Snyk vulnerability fix
# tritonclient[all]~=2.24.0
```

```
# ----- Extras
```

```
# ipython # interactive notebook
# mss # screenshots
# albumentations>=1.0.3
# pycocotools>=2.0.6 # COCO mAP
```

초기 분류 모델 성능 결과



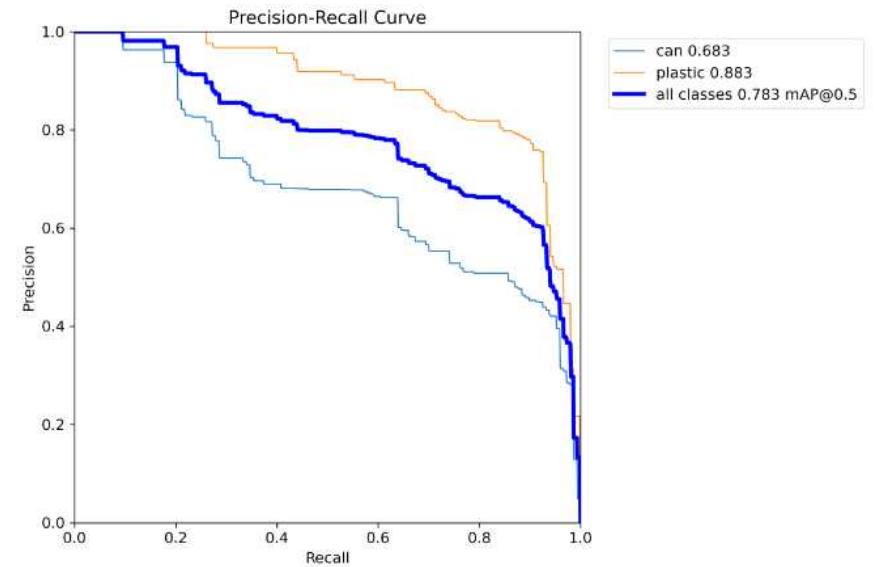
[그림 3-7] 초기 YoloV5 F1 score

F1점수는 Precision(정밀도)와 Recall(재현율)의 조화평균을 의미함.

0.0~1.0 사이의 값을 가지고 1에 가까울수록 좋은 모델임.

분류 결과 플라스틱의 경우 F1점수는 0.8이상지만, 캔의 경우 0.6의 점수에 근접함.

문제 : 데이터의 수가 부족함을 느끼고 많은 데이터를 이용해야함.



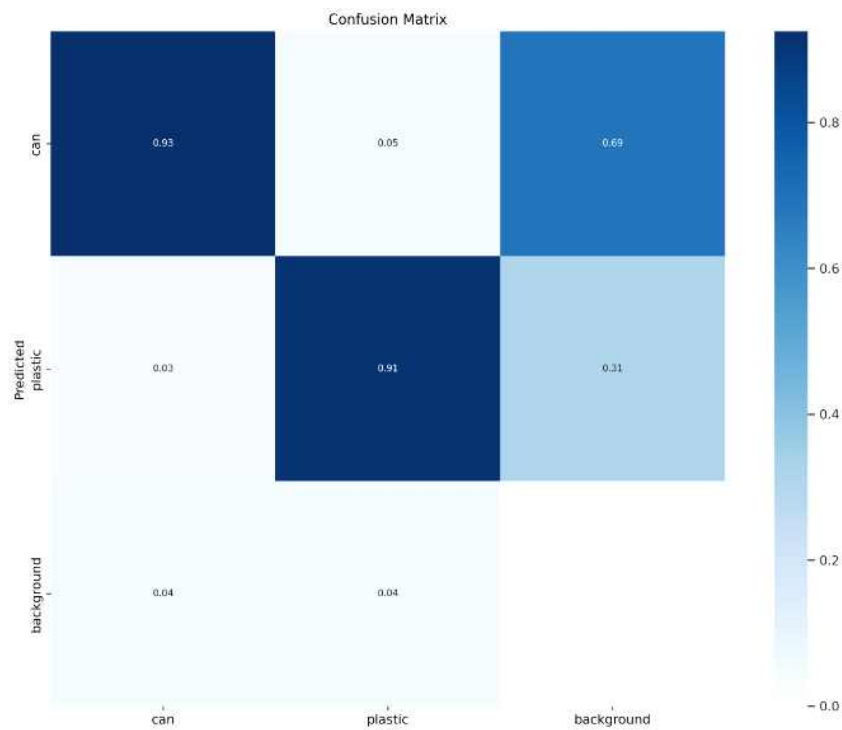
[그림 3-8] 초기 YoloV5 Precison and Recall

[그림 3-8]은 정밀도와 재현율을 그래프로 나타낸 그림임.

정밀도와 재현율은 분류성능평가지표 중 하나이며 정밀도란 모델이 True로 분류한 것 중에서 실제 True인 비율, 재현율은 실제 True인 것 중에서 모델이 True라고 예측한 것으로 정의됨.

0.0~1.0의 수치로 계산되며 1에 가까울수록 모델이 정확하게 예측한다고 생각할 수 있음.

그림으로 보았을 때 오른쪽 위 모서리와 많이 떨어져있음의 이유로 F1 점수와 동일하게 데이터 수 증가가 필요한 시점임.



[그림 3-9] 초기 YoloV5 Confusion Matrix

혼동행렬은 정밀도와 재현율을 활용하는 평가용 지수이며 모델의 성능을 평가할 때 사용되는 지표임.

위 그림을 보았을 때 실제 캔을 캔이라고 예측할 확률이 0.93%, 실제 플라스틱을 플라스틱으로 예측할 확률이 0.91%로 측정되었음.

예측할 확률이 두 클래스 모두 0.9%가 넘지만, 재현율과 정밀도를 보았을 때 좋은 성능을 가지고 있지않음으로 판단하였음.

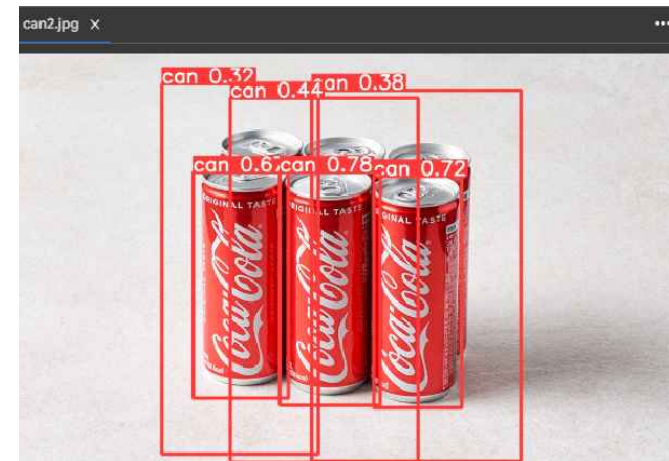
초기 YoloV5 모델 테스트 이미지 결과

테스트 이미지는 구글에서 임의로 이미지를 수집하여 테스트 하였음.



[그림 3-10] 초기 YoloV5 캔 분류 실패

철성 사이다 캔을 초기 YoloV5 모델이 테스트하였으나, 바운딩 박스가 그려지지않음. 인식 실패로 생각됨.

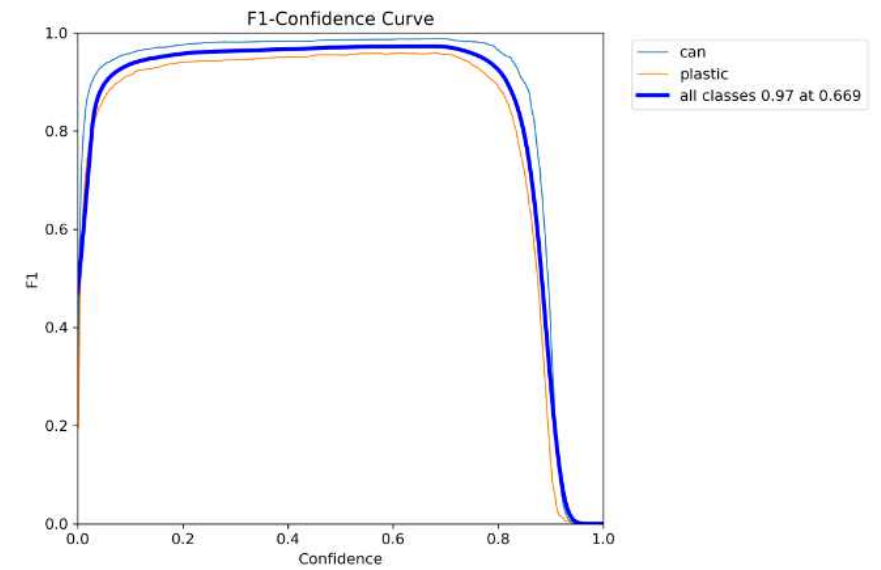


[그림 3-11] 초기 YoloV5 캔 분류 성공

다른 캔 사진은 분류에 성공하였으며 6개 캔 모두 인식하여 6개의 바운딩박스가 그려짐.



[그림 3-12] 초기 YoloV5 플라스틱 분류 성공



[그림 3-13] 데이터셋 적용 YoloV5 F1 score

플라스틱의 경우도 인식에 성공하였음.

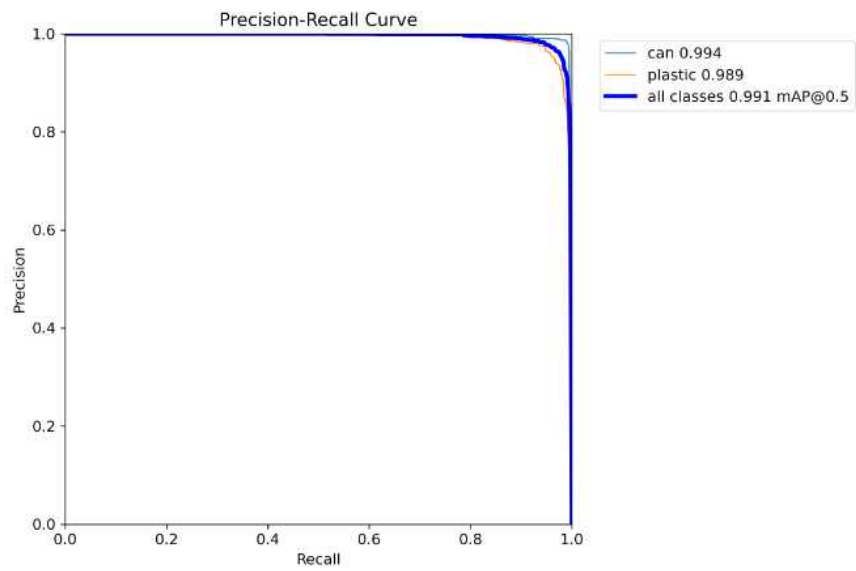
데이터셋 구축 후 분류 모델 성능 결과

2100개의 훈련데이터셋, 400개의 검증데이터셋, 전체 데이터셋 중 약 훈련세트 80%, 검증세트 20%의 비율로 훈련,검증세트를 구축하였음.

초기 YoloV5모델에서 데이터셋을 변경한 뒤, 20번의 epoch를 수행함.

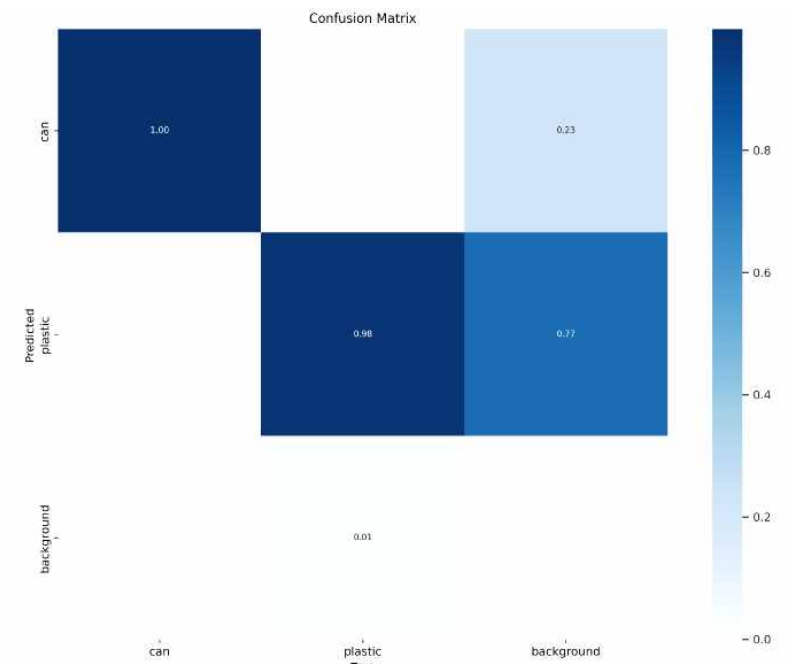
학습시간은 100분 소요됨. (1번의 epoch 당 5분가량 소요함)

초기의 그래프와 달리 부드러운 경향을 띄며, 3개의 클래스 모두 0.95 이상의 F1점수를 가짐.



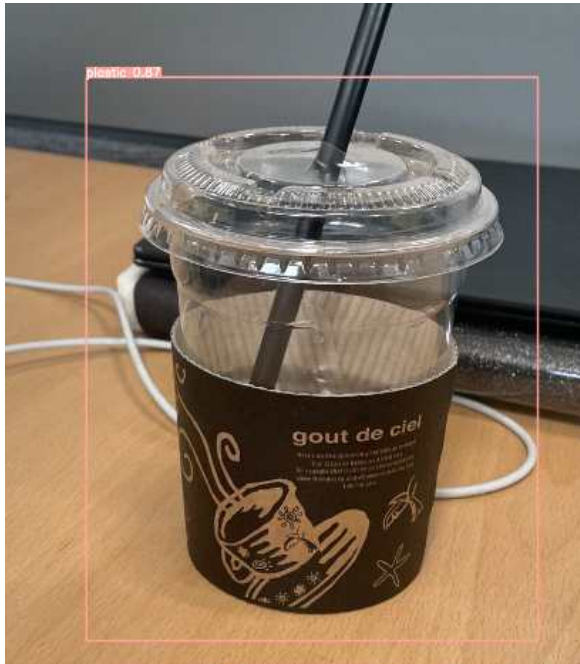
[그림 3-14] 데이터 세트 구축 후 Precision and Recall

캔의 Precision과 Recall의 점수는 0.994로 매우 높은 편에 속함. 플라스틱 또한 0.989로 매우 높은 수치를 가장.



[그림 3-15] 데이터 세트 구축 후 Confusion Matrix

혼동행렬의 결과로 캔을 캔이라 예측한 확률과 플라스틱을 플라스틱으로 예측한 확률이 매우 높게 측정되었음.



[그림 3-16] 테스트 결과

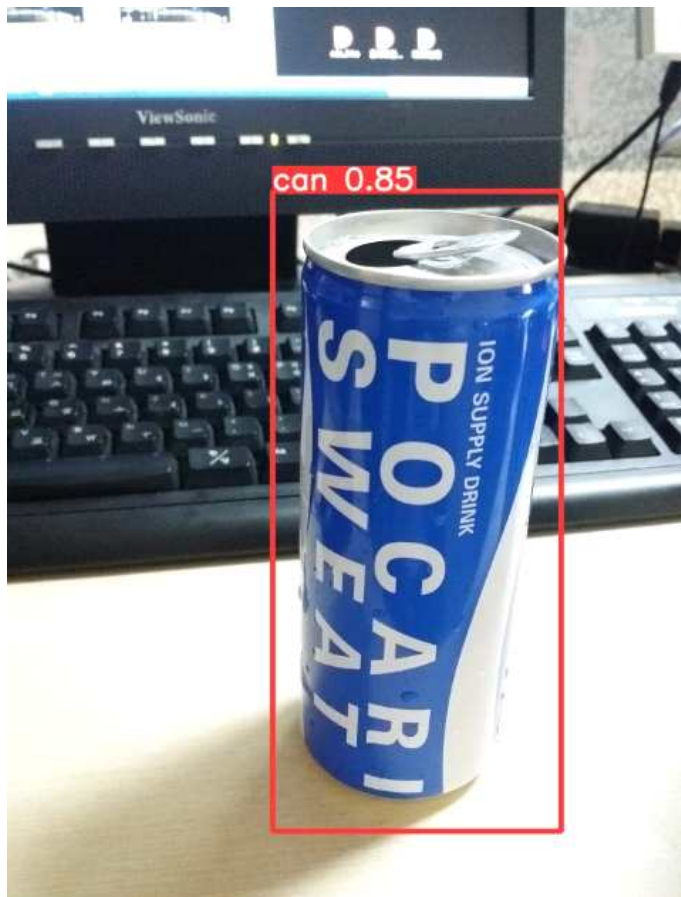
직접 찍은 사진의 테스트 결과로 일회용 플라스틱 컵을 플라스틱으로 분류에 성공하였음.



[그림 3-17] 캔 테스트 결과



[그림 3-18] 플라스틱 테스트 결과



[그림 3-19] 캔 테스트 결과



[그림 3-20] 플라스틱 테스트 결과

초기의 구축한 YoloV5 모델보다 데이터셋을 수집한 후 학습시킨 결과 캔과 플라스틱을 잘 분류하는 경향을 보임.

이후 계획으로 구축된 모델을 라즈베리파이와 카메라모듈에 적용하여 실시간으로 분류하는 과정을 계획하고있음.