YOLOv9 Training

하성욱 교수

YOLOv9의 script 기반 데이터셋 준비

```
d='./' # unzip directory
url=https://github.com/ultralytics/yolov5/releases/download/v1.0/
f='coco2017labels-segments.zip' # or 'coco2017labels.zip', 68 MB
echo 'Downloading' $url$f ' ...'
curl -L $url$f -o $f && unzip -q $f -d $d && rm $f & # download, unzip, remove in background

d='./coco/images' # unzip directory
url=http://images.cocodataset.org/zips/
f1='train2017.zip' # 19G, 118k images
f2='val2017.zip'   # 1G, 5k images
f3='test2017.zip'  # 7G, 41k images (optional)
for f in $f1 $f2 $f3; do
  echo 'Downloading' $url$f '...'
  curl -L $url$f -o $f && unzip -q $f -d $d && rm $f & # download, unzip, remove in background
done
wait
```

Coco 다운로드

- `curl -L https://github.com/ultralytics/yolov5/releases/download/v1.0/coco2017labels.zip -o coco2017labels.zip`
- `curl -L http://images.cocodataset.org/zips/train2017.zip -o train2017.zip`
- `curl -L http://images.cocodataset.org/zips/val2017.zip -o val2017.zip`
- `curl -L http://images.cocodataset.org/zips/test2017.zip -o test2017.zip`

Coco 필요한 카테고리 선정

```
# Classes
names:
  0: person
  1: bicycle
  2: car
  3: motorcycle
  4: airplane
  5: bus
  6: train
  7: truck
  8: boat
  9: traffic light
  10: fire hydrant
  11: stop sign
  12: parking meter
  13: bench
  14: bird
  15: cat
  16: dog
  17: horse
  18: sheep
  19: cow
  20: elephant
  21: bear
  22: zebra
  23: giraffe
  24: backpack
  25: umbrella
  26: handbag
  27: tie
  28: suitcase
  29: frisbee
  30: skis
  31: snowboard
  32: sports ball
  33: kite
  34: baseball bat
  35: baseball glove
  36: skateboard
  37: surfboard
  38: tennis racket
  39: bottle
  40: wine glass
  41: cup
  42: fork
  43: knife
  44: spoon
  45: bowl
  46: banana
  47: apple
```

Coco 8개 카테고리 선정하고 추출

- 0: person
- 2: car
- 5: bus
- 7: truck
- 41: cup
- 42: fork
- 43: knife
- 44: spoon

Total saved files: 79156
person = 268029
car = 45451
truck = 10384
knife = 8085
fork = 5689
cup = 21469
spoon = 6412
bus = 6344

Coco 8개 카테고리 선정하고 추출

- 24: backpack
- 25: umbrella
- 26: handbag
- 27: tie
- 41: cup
- 42: fork
- 43: knife
- 44: spoon

Total saved files: 29731
umbrella = 11672
handbag = 12882
knife = 8085
fork = 5689
cup = 21469
spoon = 6412
tie = 6700
backpack = 9085

Utility

- Val.py : 목표 카테고리만으로 데이터셋 재구성
- Spvar.py: 8:1:1(train:val:test)로 데이터 분할
- Listing.py: 텍스트 파일로 목록 저장

데이터셋 다운 후 구글 드라이브 업로드

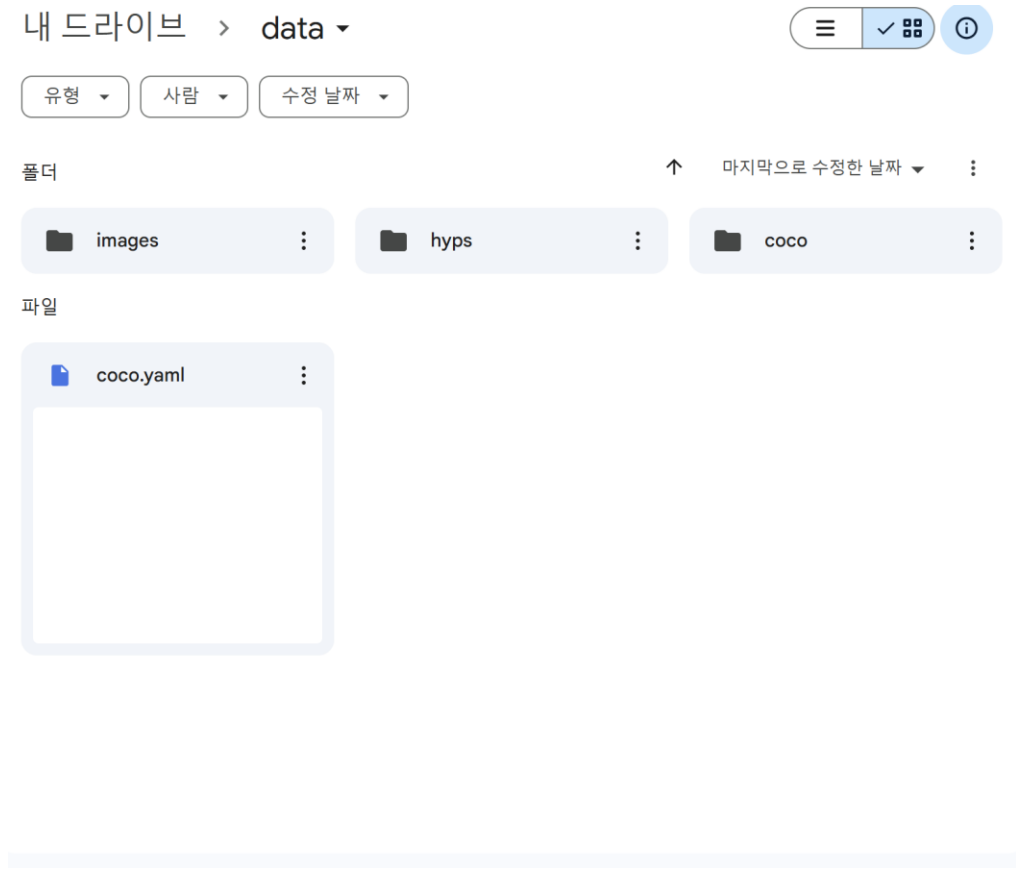
README

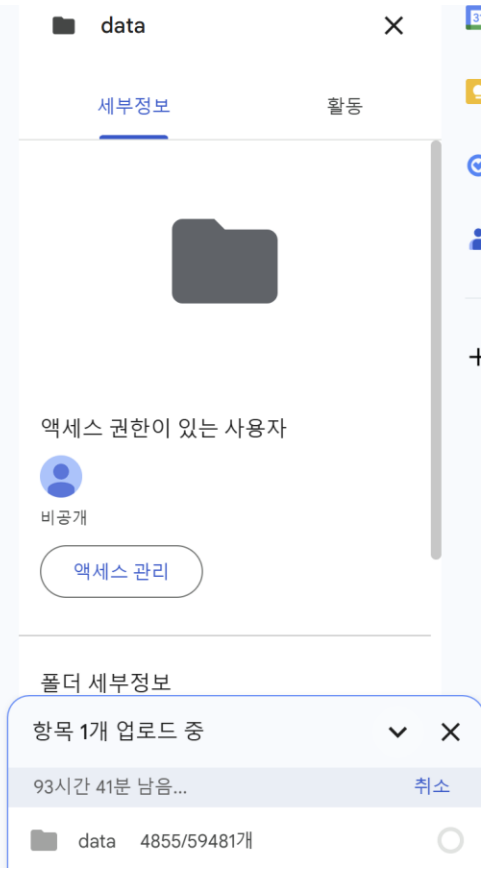
AI-Solution-IDEA

2024 AI 솔루션 아이디어 경진대회 & 창업캠프

사전 학습

데이터 셋 : 다운로드

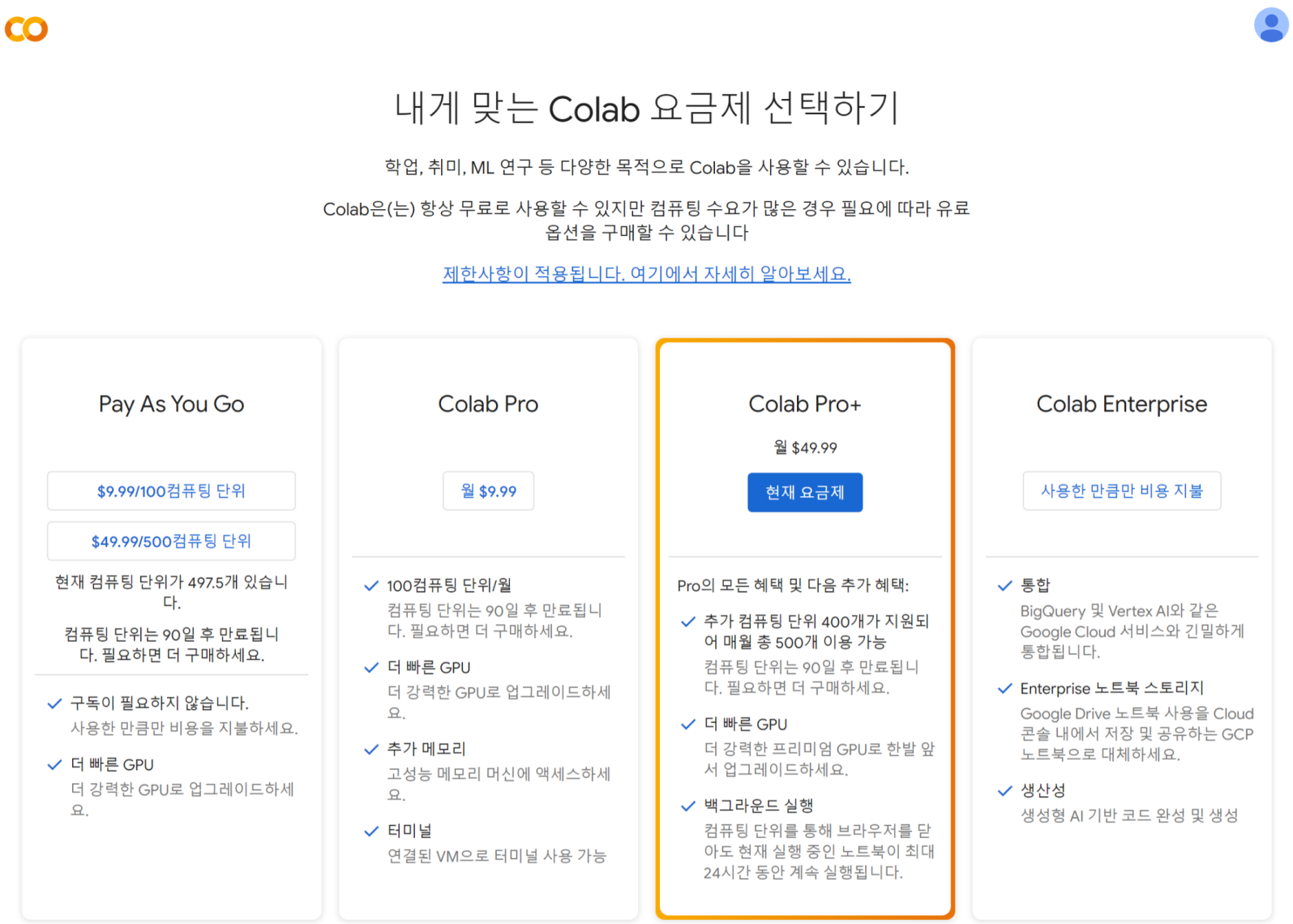




Coco.yaml 수정

```yaml
path: ../drive/MyDrive/data/coco  # dataset root dir
train: train.txt  # train images (relative to 'path') 118287 images
val: val.txt  # val images (relative to 'path') 5000 images
test: test.txt  # 20288 of 40670 images, submit to https://competitions.codalab.org/competitions/20794

# Classes
names:
  0: backpack
  1: umbrella
  2: handbag
  3: tie
  4: cup
  5: fork
  6: knife
  7: spoon
```

Colab 에서 PRO+ 업그레이드



카드 사용하고 카드 내역 확인 후

1) 대회 종료 전 접속 가능한 페이지

2) 카드 명세서

3) 본인 통장

4) 신분증 사본

5) 팀당 최대 100만원 지원

6) 반드시, 1달 후 본인이 사용해제 (추후 결제금액은 지원X)

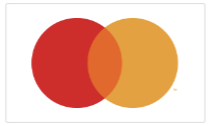
- **서류제출: fpga@yonsei.ac.kr**

결제 센터에서 인증 후 진행 필수

Google 결제 센터

거래내역 | 결제 수단 | 구독 및 서비스 | 주소 | 설정

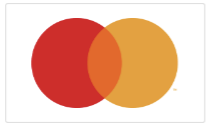
결제 수단



Mastercard ••••[redacted]

유효기간: 12/27

삭제 수정



현대 ••••[redacted]

유효기간: 11/27

삭제 수정

Colab 에서 PRO+ 확인

train_yolov9.ipynb ☆

파일 수정 보기 삽입 런타임 도구 도움말 모든 변경사항이 저장됨

댓글

+ 코드 + 텍스트

A100 RAM 디스크

먼저 GPU를 선택하기 위해서, 수정 -> 노트 설정 -> 하드웨어 가속기를 GPU로 선택한다.

[1] !nvidia-smi

1초

```
Sat Sep 28 22:01:55 2024
+---------------------------------------------------------------------------------------+
| NVIDIA-SMI 535.104.05             Driver Version: 535.104.05   CUDA Version: 12.2     |
|-----------------------------------------+----------------------+----------------------+
| GPU  Name                 Persistence-M | Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp   Perf          Pwr:Usage/Cap |         Memory-Usage | GPU-Util  Compute M. |
|                                         |                      |               MIG M. |
|=========================================+======================+======================|
|   0  NVIDIA A100-SXM4-40GB          Off | 00000000:00:04.0 Off |                    0 |
| N/A   33C    P0              46W / 400W |      2MiB / 40960MiB |      0%      Default |
|                                         |                      |             Disabled |
+-----------------------------------------+----------------------+----------------------+

+---------------------------------------------------------------------------------------+
| Processes:                                                                            |
|  GPU   GI   CI        PID   Type   Process name                            GPU Memory |
|        ID   ID                                                             Usage      |
|=======================================================================================|
|  No running processes found                                                           |
+---------------------------------------------------------------------------------------+
```

Train_yolov9

관리를 위해서 HOME에 yolov9 설정

```
[2]  import os, glob
     from IPython.display import Image
     from google.colab import drive, userdata


     HOME = os.getcwd()
     YOLO = os.path.join(HOME, 'yolov9')
     print(HOME)
     print(YOLO)
```

```
/content
/content/yolov9
```

```
[3]  # 구글 드라이브 마운트
     drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

YOLOv9 github Clone

```
!git clone https://github.com/WongKinYiu/yolov9.git
%cd yolov9
!pip install -r requirements.txt -q
```

```
Cloning into 'yolov9'...
remote: Enumerating objects: 781, done.
remote: Total 781 (delta 0), reused 0 (delta 0), pack-reused 781 (from 1)
Receiving objects: 100% (781/781), 3.27 MiB | 37.20 MiB/s, done.
Resolving deltas: 100% (331/331), done.
/content/yolov9
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 207.3/207.3 kB 14.2 MB/s eta 0
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 62.7/62.7 kB 6.1 MB/s eta 0:00
```

경량화 모델 weight 다운로드

```
[5]  !wget -P {HOME}/weights -q https://github.com/WongKinYiu/yolov9/releases/download/v0.1/gelan-c.pt
     !ls -la {HOME}/weights
```

```
total 50312
drwxr-xr-x 2 root root     4096 Sep 28 22:02 .
drwxr-xr-x 1 root root     4096 Sep 28 22:02 ..
-rw-r--r-- 1 root root 51508261 Feb 18  2024 gelan-c.pt
```

Train_yolov9

샘플 이미지 확인

```
!ls -la {HOME}/yolov9/data/images
```

```
total 140
drwxr-xr-x 2 root root   4096 Sep 28 22:02 .
drwxr-xr-x 4 root root   4096 Sep 28 22:02 ..
-rw-r--r-- 1 root root 133495 Sep 28 22:02 horses.jpg
```

c 모델로 추론 해보기

```
[7] !python detect.py --weights {HOME}/weights/gelan-c.pt --conf 0.1 --source {HOME}/yolov9/data/images/horses.jpg --device 0
```

```
detect: weights=['/content/weights/gelan-c.pt'], source=/content/yolov9/data/images/horses.jpg, data=data/coco128.yaml, imgsz=[640, 640], conf_thres=0.1, iou
YOLO 🚀 v0.1-104-g5b1ea9a Python-3.10.12 torch-2.4.1+cu121 CUDA:0 (NVIDIA A100-SXM4-40GB, 40514MiB)

/content/yolov9/models/experimental.py:243: FutureWarning: You are using `torch.load` with `weights_only=False` (the current default value), which uses the de
  ckpt = torch.load(attempt_download(w), map_location='cpu')  # load
Fusing layers...
Model summary: 387 layers, 25288768 parameters, 0 gradients, 102.1 GFLOPs
image 1/1 /content/yolov9/data/images/horses.jpg: 448x640 5 horses, 64.9ms
Speed: 0.5ms pre-process, 64.9ms inference, 726.2ms NMS per image at shape (1, 3, 640, 640)
Results saved to runs/detect/exp
```

Train_yolov9

0초

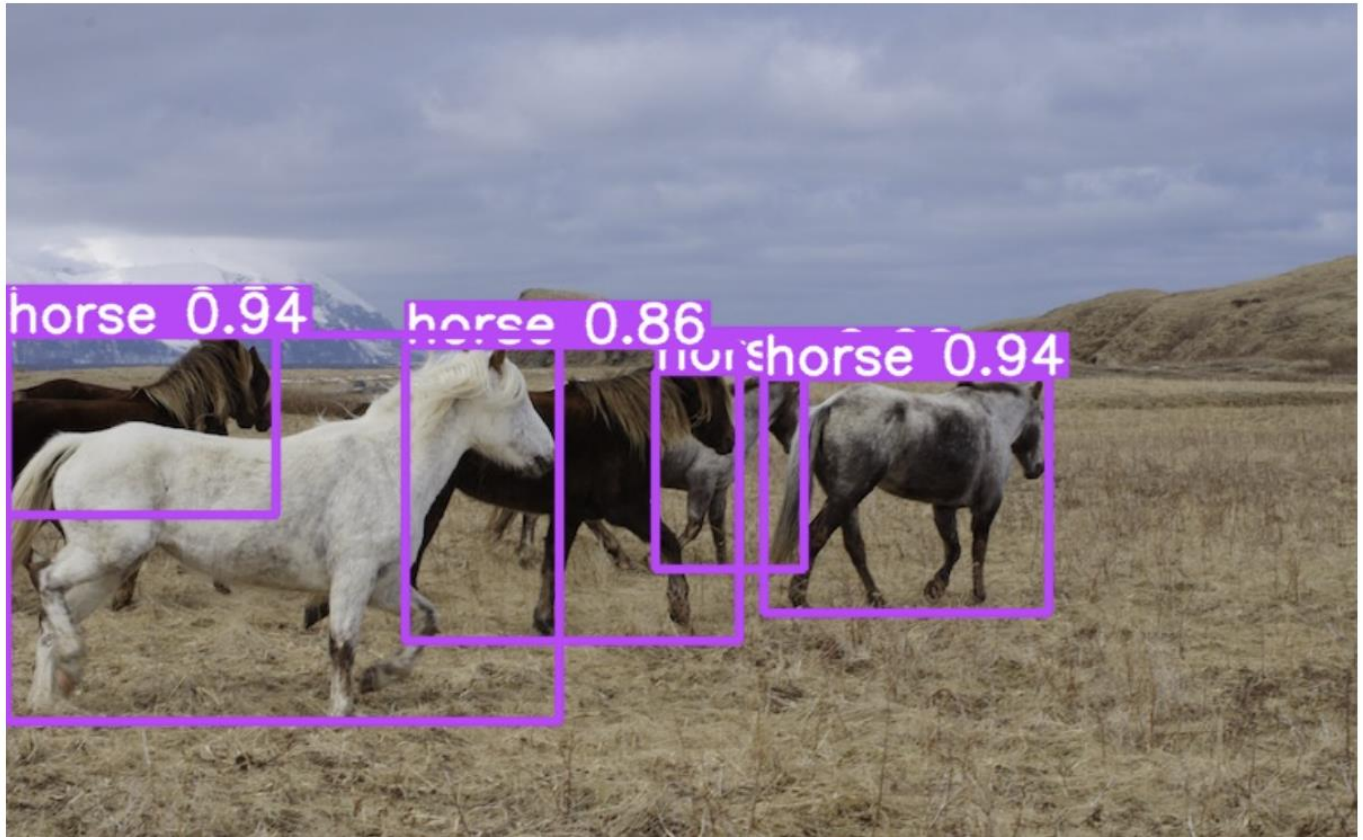
```
!ls -la {HOME}/yolov9/runs/detect
```

```
total 12
drwxr-xr-x 3 root root 4096 Sep 28 22:02 .
drwxr-xr-x 3 root root 4096 Sep 28 22:02 ..
drwxr-xr-x 2 root root 4096 Sep 28 22:02 exp
```

1초

[9]

```
from IPython.display import Image

Image(filename=f"{HOME}/yolov9/runs/detect/exp/horses.jpg", width=600)
```



Train_yolov9

커스텀 coco 데이터셋 학습

```
%cd {HOME}/yolov9

!python train.py ₩
--batch 16 --epochs 25 --img 640 --device 0 --min-items 0 --close-mosaic 15 ₩
--data /content/drive/MyDrive/data/coco.yaml ₩
--weights {HOME}/weights/gelan-c.pt ₩
--cfg models/detect/gelan-c.yaml ₩
--hyp hyp.scratch-high.yaml
```

```
/content/yolov9
2024-09-28 22:03:20.032309: I tensorflow/core/util/port.cc:153] oneDNN custom operations are on. You may see slightly different numerical results due to floating-point rou
2024-09-28 22:03:20.050055: E external/local_xla/xla/stream_executor/cuda/cuda_fft.cc:485] Unable to register cuFFT factory: Attempting to register factory for plugin cuFF
2024-09-28 22:03:20.071335: E external/local_xla/xla/stream_executor/cuda/cuda_dnn.cc:8454] Unable to register cuDNN factory: Attempting to register factory for plugin cuD
2024-09-28 22:03:20.077751: E external/local_xla/xla/stream_executor/cuda/cuda_blas.cc:1452] Unable to register cuBLAS factory: Attempting to register factory for plugin c
2024-09-28 22:03:20.093191: I tensorflow/core/platform/cpu_feature_guard.cc:210] This TensorFlow binary is optimized to use available CPU instructions in performance-criti
To enable the following instructions: AVX2 AVX512F AVX512_VNNI FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
2024-09-28 22:03:21.309424: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-TRT Warning: Could not find TensorRT
train: weights=/content/weights/gelan-c.pt, cfg=models/detect/gelan-c.yaml, data=/content/drive/MyDrive/data/coco.yaml, hyp=hyp.scratch-high.yaml, epochs=25, batch_size=16
YOLO 🚀 v0.1-104-g5b1ea9a Python-3.10.12 torch-2.4.1+cu121 CUDA:0 (NVIDIA A100-SXM4-40GB, 40514MiB)

hyperparameters: lr0=0.01, lrf=0.01, momentum=0.937, weight_decay=0.0005, warmup_epochs=3.0, warmup_momentum=0.8, warmup_bias_lr=0.1, box=7.5, cls=0.5, cls_pw=1.0, obj=0.
ClearML: run 'pip install clearml' to automatically track, visualize and remotely train YOLO 🚀 in ClearML
Comet: run 'pip install comet_ml' to automatically track and visualize YOLO 🚀 runs in Comet
TensorBoard: Start with 'tensorboard --logdir runs/train', view at http://localhost:6006/
Downloading https://ultralytics.com/assets/Arial.ttf to /root/.config/Ultralytics/Arial.ttf...
100% 755k/755k [00:00<00:00, 64.2MB/s]
/content/yolov9/train.py:108: FutureWarning: You are using `torch.load` with `weights_only=False` (the current default value), which uses the default pickle module implici
  ckpt = torch.load(weights, map_location='cpu')  # load checkpoint to CPU to avoid CUDA memory leak
Overriding model.yaml nc=80 with nc=8
```

Train_yolov9

```
      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
      21/24      14.9G      1.142      1.302      1.188         28        640: 100% 1487/1487 [04:53<00:00,  5.06it/s]
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 93/93 [00:26<00:00,  3.52it/s]
                   all       2973       8015      0.618      0.508      0.554      0.379

      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
      22/24      14.9G      1.121      1.259      1.172         22        640: 100% 1487/1487 [04:53<00:00,  5.06it/s]
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 93/93 [00:26<00:00,  3.57it/s]
                   all       2973       8015      0.612      0.522      0.559      0.383

      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
      23/24      14.9G      1.112      1.221      1.164         10        640: 100% 1487/1487 [04:54<00:00,  5.06it/s]
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 93/93 [00:26<00:00,  3.57it/s]
                   all       2973       8015      0.633      0.514      0.563      0.389

      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
      24/24      14.9G       1.08      1.172      1.149         30        640: 100% 1487/1487 [04:54<00:00,  5.05it/s]
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 93/93 [00:25<00:00,  3.61it/s]
                   all       2973       8015       0.65      0.508      0.567      0.392

25 epochs completed in 2.335 hours.
```

Train_yolov9

```
Traceback (most recent call last):
  File "/usr/lib/python3.10/threading.py", line 1016, in _bootstrap_inner
    self.run()
  File "/usr/lib/python3.10/threading.py", line 953, in run
    self._target(*self._args, **self._kwargs)
  File "/content/yolov9/utils/plots.py", line 300, in plot_images
    annotator.box_label(box, label, color=color)
  File "/content/yolov9/utils/plots.py", line 86, in box_label
    w, h = self.font.getsize(label)  # text width, height
AttributeError: 'FreeTypeFont' object has no attribute 'getsize'
```

Class	Images	Instances	P	R	mAP50	mAP50-95: 100% 93/93 [00:28<00:00, 3.26it/s]
all	2973	8015	0.648	0.509	0.567	0.393
backpack	2973	912	0.566	0.381	0.411	0.236
umbrella	2973	1075	0.661	0.647	0.685	0.477
handbag	2973	1315	0.567	0.344	0.41	0.247
tie	2973	683	0.722	0.624	0.696	0.482
cup	2973	2102	0.72	0.638	0.714	0.54
fork	2973	582	0.72	0.572	0.636	0.473
knife	2973	752	0.606	0.456	0.511	0.335
spoon	2973	594	0.62	0.409	0.475	0.35

Results saved to **runs/train/exp**

Train_yolov9

학습 결과

디폴트로 {HOME}/yolov9/runs/train/에 exp, exp2, exp3, ... 순서로 1씩 증가된 값으로 저장된다. 마지막 exp?가 최근 학습한 모델이다. --name 파라미터를 사용해서 덮어쓸 수 도 있다.

```
[11] !ls {HOME}/yolov9/runs/train
```

```
exp
```

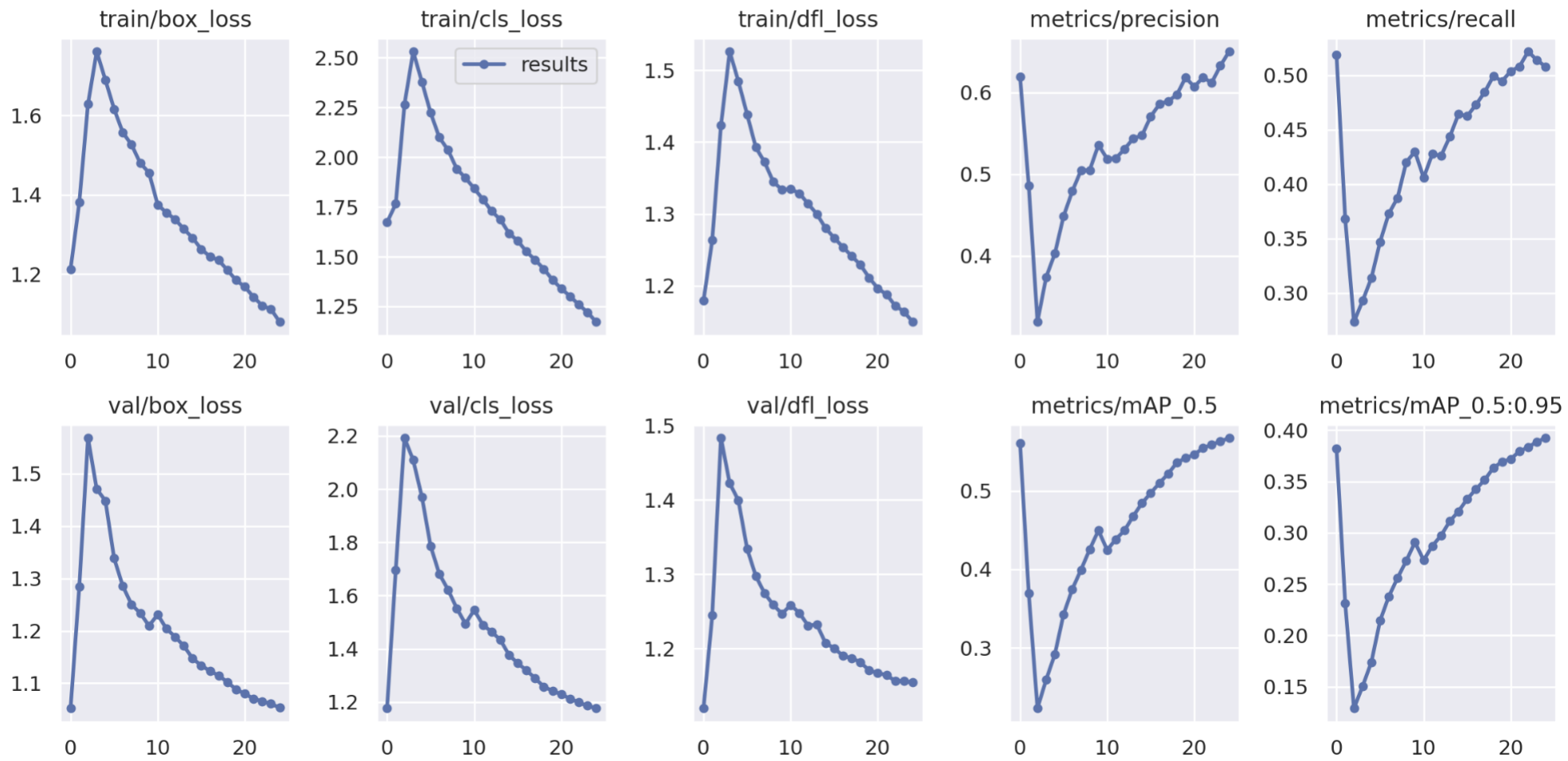
```
!ls {HOME}/yolov9/runs/train/exp/
```

```
confusion_matrix.png                                labels_correlogram.jpg  PR_curve.png  weights
events.out.tfevents.1727561002.4e251f1077dd.1665.0  labels.jpg              R_curve.png
F1_curve.png                                        opt.yaml                results.csv
hyp.yaml                                            P_curve.png             results.png
```

Train yolov9

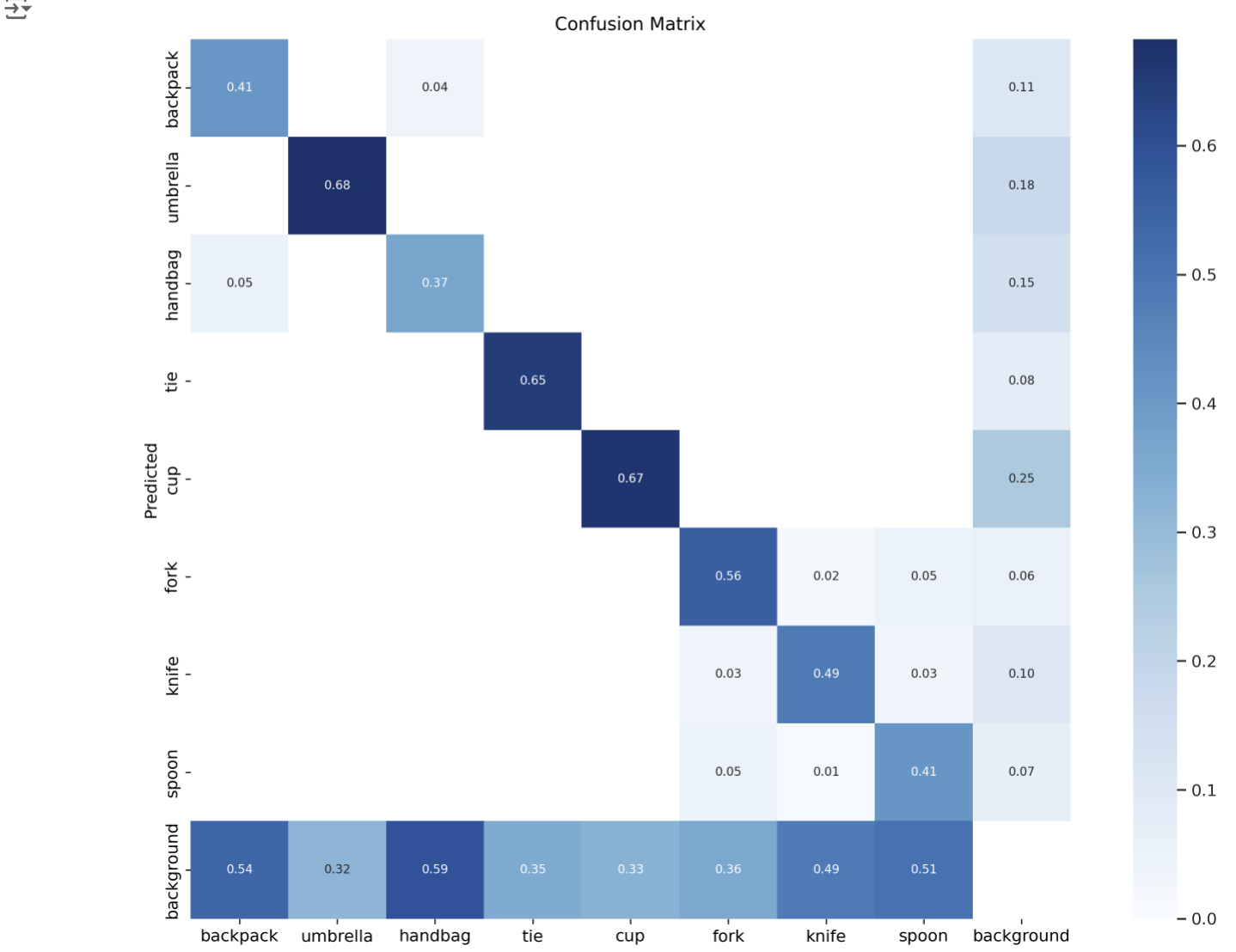
1초

```
from IPython.display import Image

Image(filename=f"{HOME}/yolov9/runs/train/exp/results.png", width=1000)
```

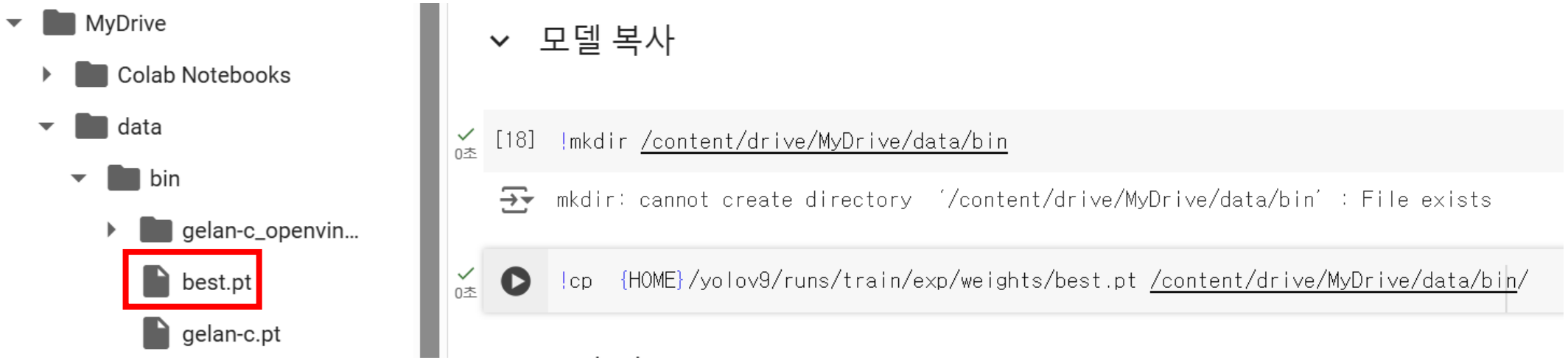


Train_yolov9

```
from IPython.display import Image

Image(filename=f"{HOME}/yolov9/runs/train/exp/confusion_matrix.png", width=1000)
```



Train_yolov9



Train_yolov9

모델 테스트

```
!python detect.py ₩
--img 1280 --conf 0.1 --device 0 ₩
--weights {HOME}/yolov9/runs/train/exp/weights/best.pt ₩
--source /content/drive/MyDrive/data/coco/images/test/
```

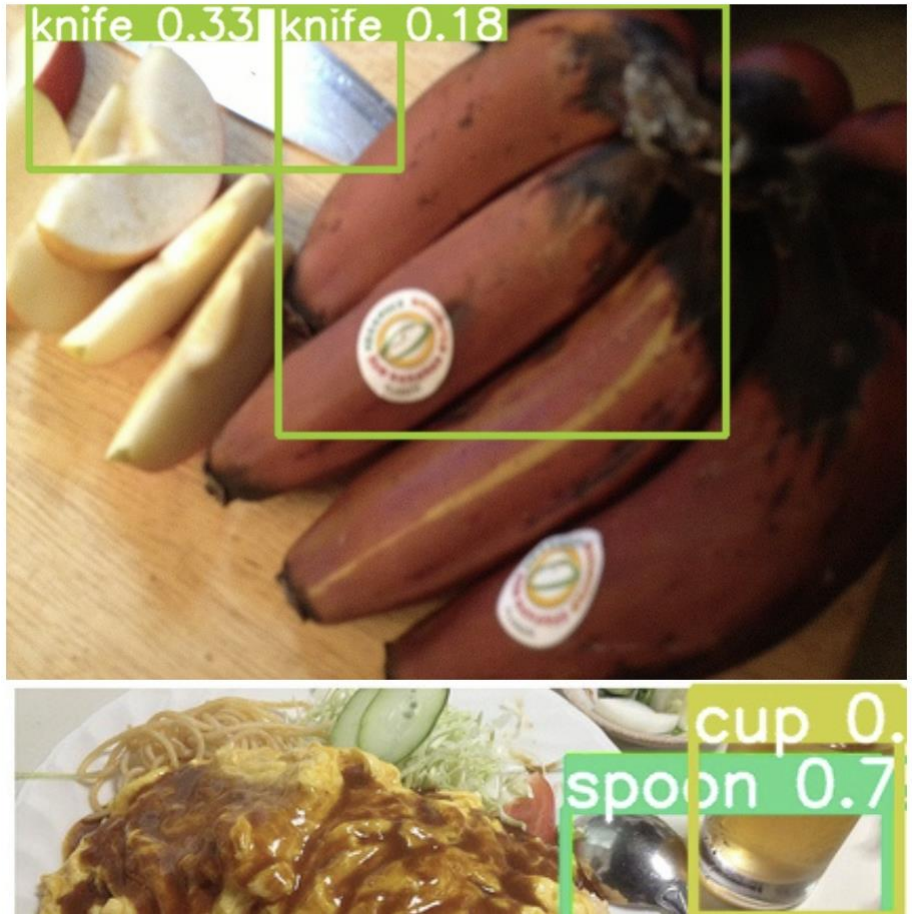
```
detect: weights=['/content/yolov9/runs/train/exp/weights/best.pt'], source=/content/drive/MyDrive/data/coco/images/test/, data=data/coc
YOLO 🚀 v0.1-104-g5b1ea9a Python-3.10.12 torch-2.4.1+cu121 CUDA:0 (NVIDIA A100-SXM4-40GB, 40514MiB)

/content/yolov9/models/experimental.py:243: FutureWarning: You are using `torch.load` with `weights_only=False` (the current default val
  ckpt = torch.load(attempt_download(w), map_location='cpu')  # load
Fusing layers...
gelan-c summary: 387 layers, 25233256 parameters, 0 gradients, 101.8 GFLOPs
image 1/2974 /content/drive/MyDrive/data/coco/images/test/000000524450.jpg: 864x1280 2 backpacks, 2 handbags, 68.0ms
image 2/2974 /content/drive/MyDrive/data/coco/images/test/000000524456.jpg: 960x1280 1 handbag, 2 ties, 1 knife, 69.7ms
image 3/2974 /content/drive/MyDrive/data/coco/images/test/000000524470.jpg: 1280x1280 1 backpack, 17.5ms
image 4/2974 /content/drive/MyDrive/data/coco/images/test/000000524507.jpg: 864x1280 2 umbrellas, 1 handbag, 16.2ms
image 5/2974 /content/drive/MyDrive/data/coco/images/test/000000524518.jpg: 1280x960 7 cups, 1 fork, 2 knifes, 65.9ms
image 6/2974 /content/drive/MyDrive/data/coco/images/test/000000524536.jpg: 1280x864 1 backpack, 2 umbrellas, 1 handbag, 1 tie, 67.0ms
image 7/2974 /content/drive/MyDrive/data/coco/images/test/000000524623.jpg: 1280x992 2 backpacks, 67.6ms
image 8/2974 /content/drive/MyDrive/data/coco/images/test/000000524645.jpg: 960x1280 2 cups, 1 fork, 3 knifes, 16.7ms
image 9/2974 /content/drive/MyDrive/data/coco/images/test/000000524651.jpg: 960x1280 5 backpacks, 15.7ms
image 10/2974 /content/drive/MyDrive/data/coco/images/test/000000524656.jpg: 960x1280 21 cups, 2 knifes, 2 spoons, 15.6ms
image 11/2974 /content/drive/MyDrive/data/coco/images/test/000000524665.jpg: 1280x864 2 backpacks, 1 handbag, 16.3ms
image 12/2974 /content/drive/MyDrive/data/coco/images/test/000000524690.jpg: 992x1280 1 fork, 67.4ms
image 13/2974 /content/drive/MyDrive/data/coco/images/test/000000524695.jpg: 960x1280 2 cups, 16.2ms
image 14/2974 /content/drive/MyDrive/data/coco/images/test/000000524710.jpg: 1280x1280 2 cups, 4 spoons, 17.5ms
image 15/2974 /content/drive/MyDrive/data/coco/images/test/000000524736.jpg: 928x1280 (no detections), 67.6ms
image 16/2974 /content/drive/MyDrive/data/coco/images/test/000000524766.jpg: 928x1280 1 handbag, 2 ties, 1 cup, 15.7ms
image 17/2974 /content/drive/MyDrive/data/coco/images/test/000000524775.jpg: 736x1280 3 backpacks, 1 handbag, 68.1ms
...
image 2971/2974 /content/drive/MyDrive/data/coco/images/test/000000581829.jpg: 864x1280 1 spoon, 15.1ms
image 2972/2974 /content/drive/MyDrive/data/coco/images/test/000000581857.jpg: 1280x864 1 umbrella, 7 handbags, 15.8ms
image 2973/2974 /content/drive/MyDrive/data/coco/images/test/000000581900.jpg: 1056x1280 1 handbag, 15.1ms
image 2974/2974 /content/drive/MyDrive/data/coco/images/test/000000581904.jpg: 864x1280 4 backpacks, 5 handbags, 15.2ms
Speed: 1.3ms pre-process, 16.2ms inference, 1.2ms NMS per image at shape (1, 3, 1280, 1280)
Results saved to runs/detect/exp2
```

Train_yolov9

```
[21] !ls {HOME}/yolov9/runs/detect
```

```
exp  exp2
```

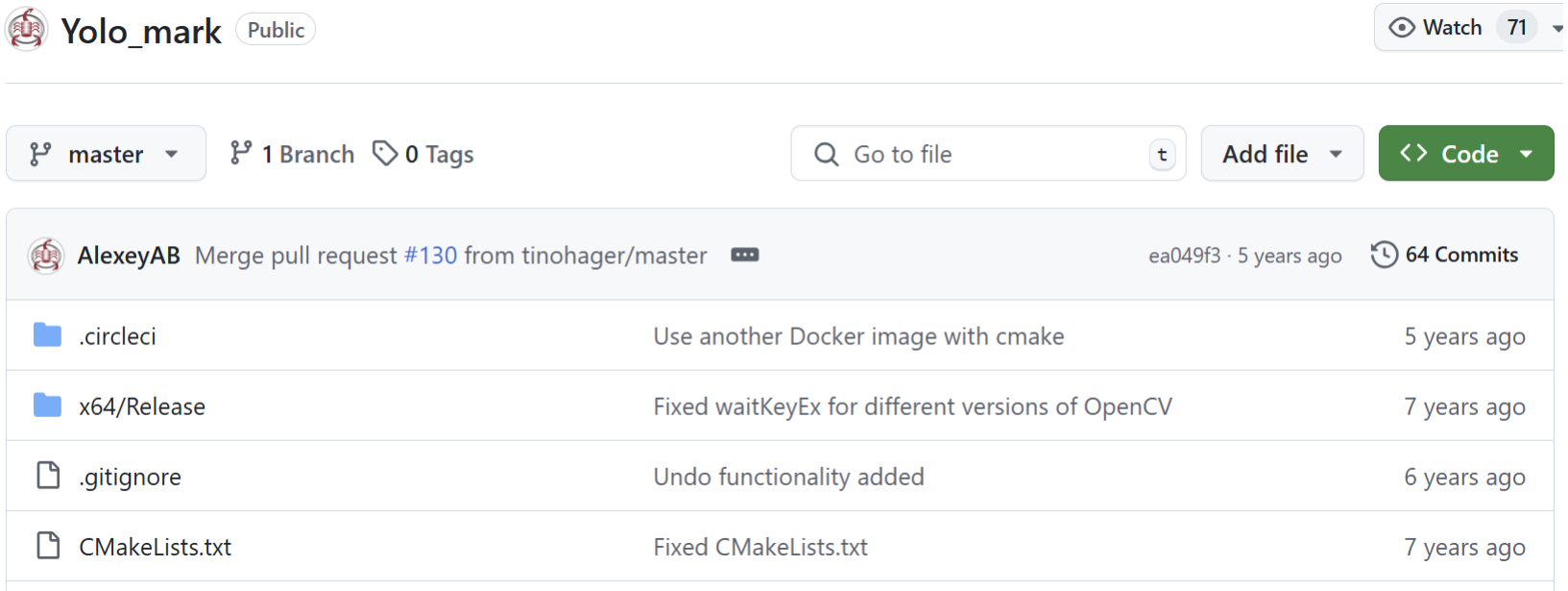
```
import glob

from IPython.display import Image, display

for image_path in glob.glob(f'{HOME}/yolov9/runs/detect/exp2/*.jpg')[:2]:
      display(Image(filename=image_path, width=600))
```



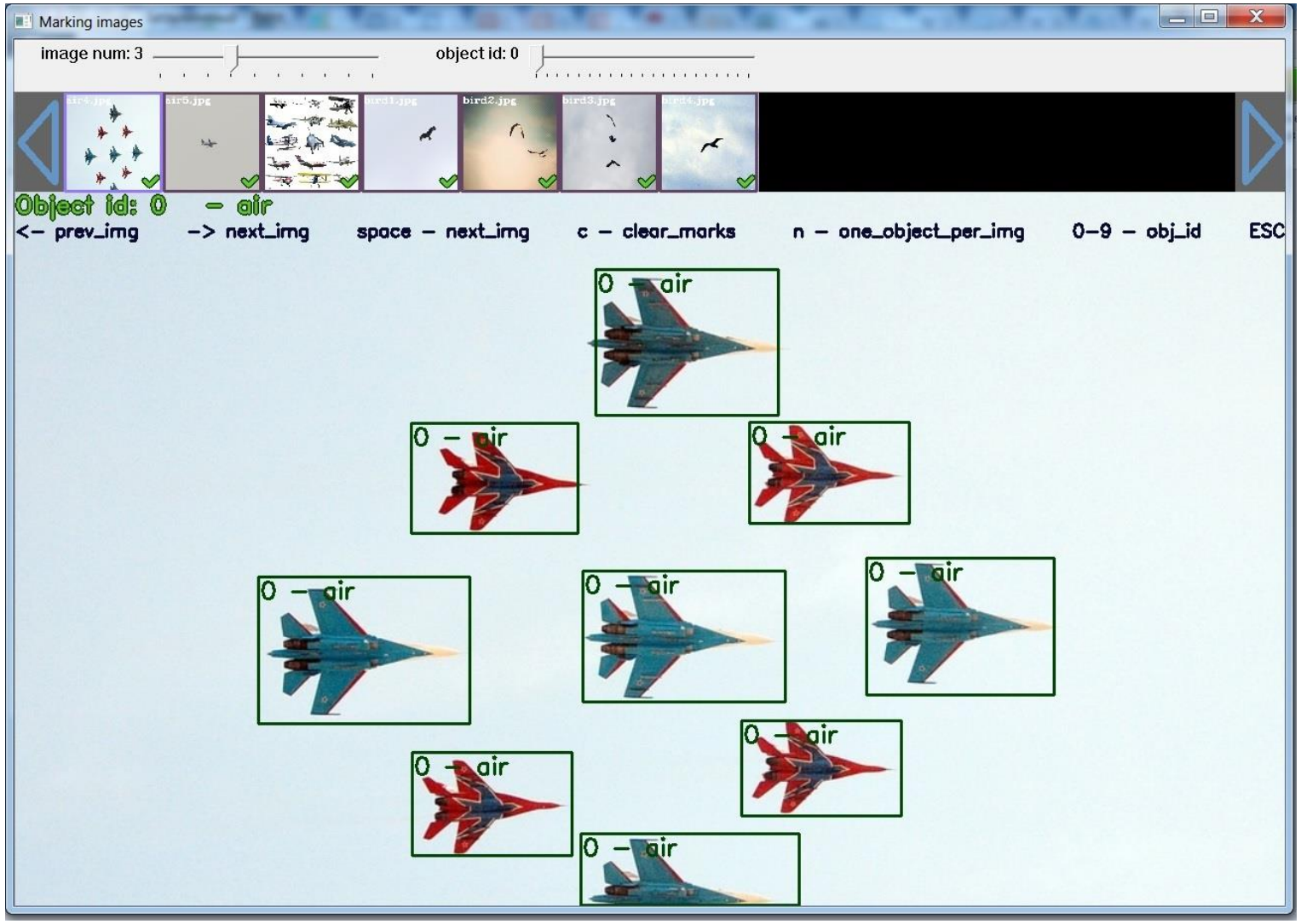
실제 데이터셋 구축

- 카메라 (CCTV, 폰 카메라, 노트북 캠, 등)
- 드론 카메라 등으로 수집

- Yolomark로 라벨링(https://github.com/AlexeyAB/Yolo_mark)



실제 데이터셋 라벨링



Mouse control

Button	Description
Left	Draw box
Right	Move box

Keyboard Shortcuts

Shortcut	Description
→	Next image
←	Previous image
r	Delete selected box (mouse hovered)
c	Clear all marks on the current image
p	Copy previous mark
o	Track objects
ESC	Close application
n	One object per image
0-9	Object id
m	Show coords
w	Line width
k	Hide object name
h	Help