

지능화 캡스톤 프로젝트

프로젝트 #2 중간 발표

2022. 5. 25

충북대학교 산업인공지능학과

21-3조, 최준혁, 이지연

데이터 라벨링 진행

README.rst

Labellmg

pypi v1.8.6 build passing lang en lang zh
lang zh-TW

Labellmg is a graphical image annotation tool.

It is written in Python and uses Qt for its graphical interface.

Annotations are saved as XML files in PASCAL VOC format, the format used by [ImageNet](#). Besides, it also supports YOLO and CreateML formats.



Labellmg 툴 사용.

Kaggle 제공 데이터 사용.

Safety Helmet Detection

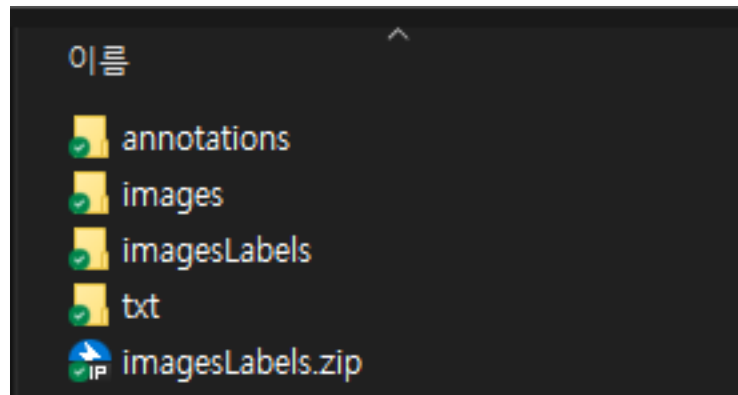
Improve work safety by detecting the presence of people and safety helmets.

Data Code (12) Discussion (6) Metadata

About Dataset



데이터 라벨링 진행

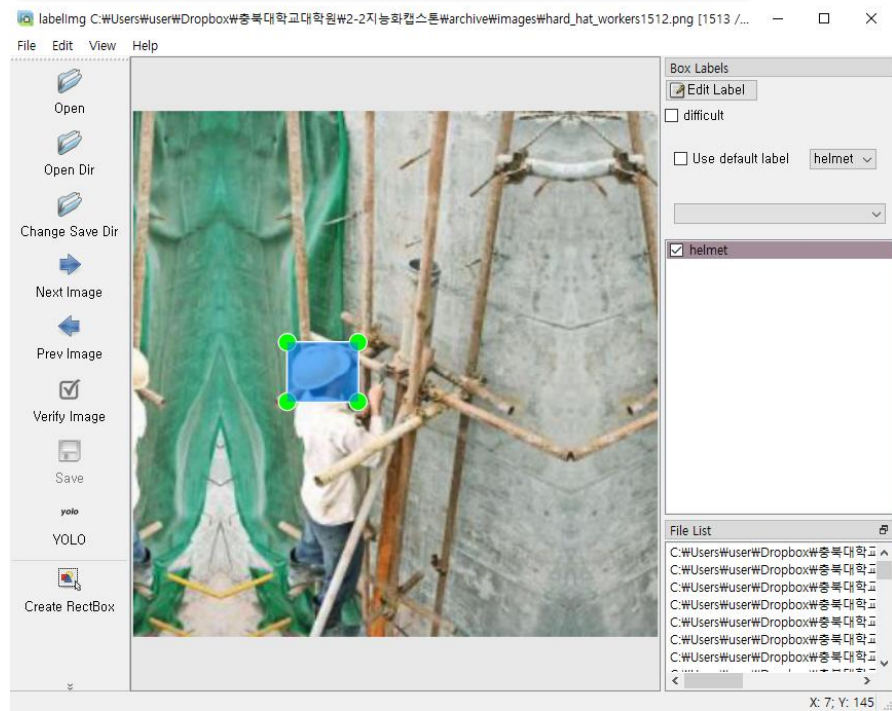


Annotations : xml 데이터

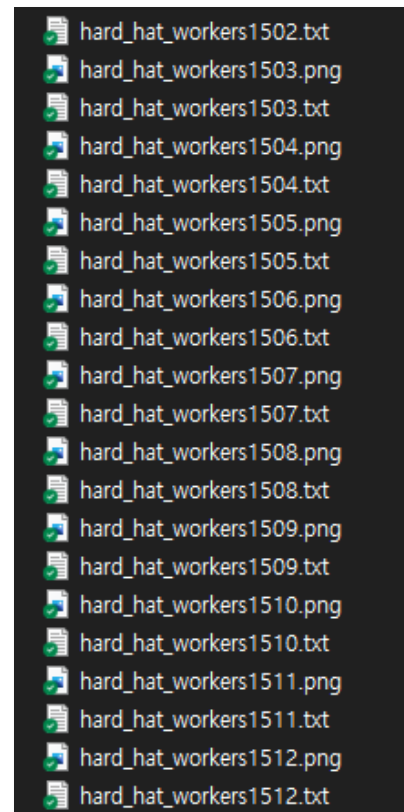
Images : png 데이터

imagesLabels : 라벨링 완료된 이미지와 txt 파일 한 폴더에 모아둠

txt : yolo 학습을 위한 라벨링 텍스트 데이터



라벨링 진행 툴



약 1500개 진행



너무 작아서 구분이
육안으로 어려운 헬멧은
라벨링 진행X

YOLOv5 모델학습

```
[5] #데이터 확인
#cd /content/yolo5/
#lsat /content/dataset/data.yaai

#cd /content/yolo5/
#lsat /content/drive/MyDrive/dataset/data.yaai

/content/yolo5
train: ../train/images
val: ../valid/images

nc: 2
names: ['helmet', 'head']

[6] %cd /
from glob import glob

#img_list = glob('/content/dataset/export/images/*.png')
img_list = glob('/content/drive/MyDrive/dataset/images/*.png')
print(len(img_list))

/
1481

[7] # train, validation 분리
from sklearn.model_selection import train_test_split
train_img_list, val_img_list = train_test_split(img_list, test_size=0.2, random_state=2000)
print(len(train_img_list), len(val_img_list))

1184 297

[8] #train/val 이미지 경로 저장.
with open('/content/train.txt', 'w') as f:
    f.write('\n'.join(train_img_list) + '\n')

with open('/content/val.txt', 'w') as f:
    f.write('\n'.join(val_img_list) + '\n')
```

```
[10] %cd /content/yolo5/

#python train.py --img 416 --batch 16 --epochs 50 --data /content/dataset/data.yaai --cfg /models/yolo5s.yaml --weights yolo5s.pt --name gun_yolo5s_results
!python train.py --img 416 --batch 16 --epochs 50 --data /content/drive/MyDrive/dataset/data.yaai --cfg /models/yolo5s.yaml --weights yolo5s.pt --name gun_yolo5s_results

Epoch 41/49   box       obj       cls     labels  img_size
              1.936   0.02966  0.01627  0.001124  62
              Class  Images  Labels  P         R         mAPv.5  mAPv.5: 95: 100x 10/10 [00:04:00:00, 2.39it/s]
              all    297      970     0.933     0.876     0.944     0.539

Epoch 42/49   box       obj       cls     labels  img_size
              1.936   0.03051  0.01637  0.001075  74
              Class  Images  Labels  P         R         mAPv.5  mAPv.5: 95: 100x 10/10 [00:04:00:00, 2.47it/s]
              all    297      970     0.919     0.888     0.943     0.54

Epoch 43/49   box       obj       cls     labels  img_size
              1.936   0.02983  0.0163  0.001005  53
              Class  Images  Labels  P         R         mAPv.5  mAPv.5: 95: 100x 10/10 [00:03:00:00, 2.27it/s]
              all    297      970     0.915     0.893     0.942     0.541

Epoch 44/49   box       obj       cls     labels  img_size
              1.936   0.03002  0.01573  0.00102   53
              Class  Images  Labels  P         R         mAPv.5  mAPv.5: 95: 100x 10/10 [00:04:00:00, 2.41it/s]
              all    297      970     0.929     0.872     0.94   0.542

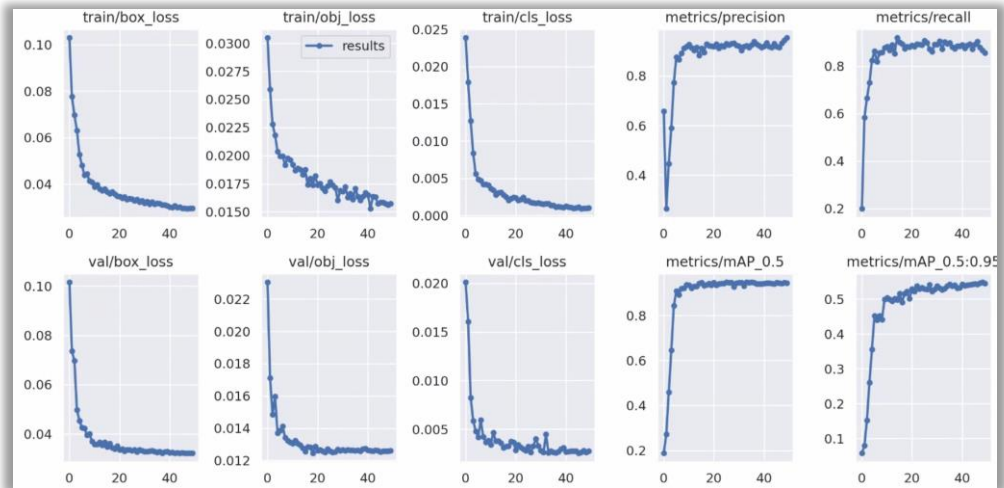
Epoch 45/49   box       obj       cls     labels  img_size
              1.936   0.02959  0.01583  0.001144  66
              Class  Images  Labels  P         R         mAPv.5  mAPv.5: 95: 100x 10/10 [00:04:00:00, 2.44it/s]
              all    297      970     0.918     0.892     0.945     0.544

Epoch 46/49   box       obj       cls     labels  img_size
              1.936   0.0296   0.01584  0.000913  67
              Class  Images  Labels  P         R         mAPv.5  mAPv.5: 95: 100x 10/10 [00:03:00:00, 2.28it/s]
              all    297      970     0.914     0.903     0.944     0.542

Epoch 47/49   box       obj       cls     labels  img_size
              1.936   0.02929  0.01574  0.000964  82
              Class  Images  Labels  P         R         mAPv.5  mAPv.5: 95: 100x 10/10 [00:03:00:00, 2.25it/s]
              all    297      970     0.932     0.879     0.942     0.547

Epoch 48/49   box       obj       cls     labels  img_size
              1.936   0.02948  0.01562  0.000959  88
              Class  Images  Labels  P         R         mAPv.5  mAPv.5: 95: 100x 10/10 [00:04:00:00, 2.42it/s]
              all    297      970     0.943     0.867     0.945     0.549

Epoch 49/49   box       obj       cls     labels  img_size
              1.936   0.0296   0.01574  0.001026  95
              Class  Images  Labels  P         R         mAPv.5  mAPv.5: 95: 100x 74/74 [00:32:00:00, 2.26it/s]
```



데이터 분리

Train.py를 통해 학습
(Batch: 16, Epoch: 50)

구글드라이브에 마운트하여 Colab에서 진행.

학습결과

YOLOv5 모델학습

```
val_img_path = val_img_list[1]

#!python detect.py --weights /content/yolov5/runs/train/gun_yolov5s_results/weights/best.pt --img 416 --conf 0.5 --source "{val_img_path}"

!python detect.py --weights /content/yolov5/runs/train/gun_yolov5s_results/weights/best.pt --img 416 --conf 0.5 --source /content/hard_hat_workers4937.png

detect: weights=['/content/yolov5/runs/train/gun_yolov5s_results/weights/best.pt'], source=/content/hard_hat_workers4937.png, data=data/coco128.yaml, imgs=2
YOLOv5 v6.1-220-g68ff6c9 Python-3.7.13 torch-1.11.0+cu113 CUDA:0 (Tesla T4, 15110MiB)

Fusing layers...
YOLOv5s summary: 213 layers, 7015519 parameters, 0 gradients, 15.8 GFLOPs
image 1/1 /content/hard_hat_workers4937.png: 416x416 1 helmet, 7 heads, Done. (0.009s)
Speed: 0.4ms pre-process, 9.5ms inference, 1.2ms NMS per image at shape (1, 3, 416, 416)
Results saved to runs/detect/exp
```

Best.pt을 이용해 테스트.
(캐글 데이터 중 사용하지 않은 데이터를 이용)



image4937



image4890

- Image4937 에서 착용하지 않은 헬멧이 검출됨.
- Image4890 에서 검출되지 않은 항목 존재.

이후 진행방향

1. 헬멧 / 헤드 데이터 추가.
 - Kaggle에서 사용하지 않은 데이터 및 검색을 통해 데이터 확보.
2. 정확도 향상을 위한 작업.

감사합니다.