

신호등 분류 모델

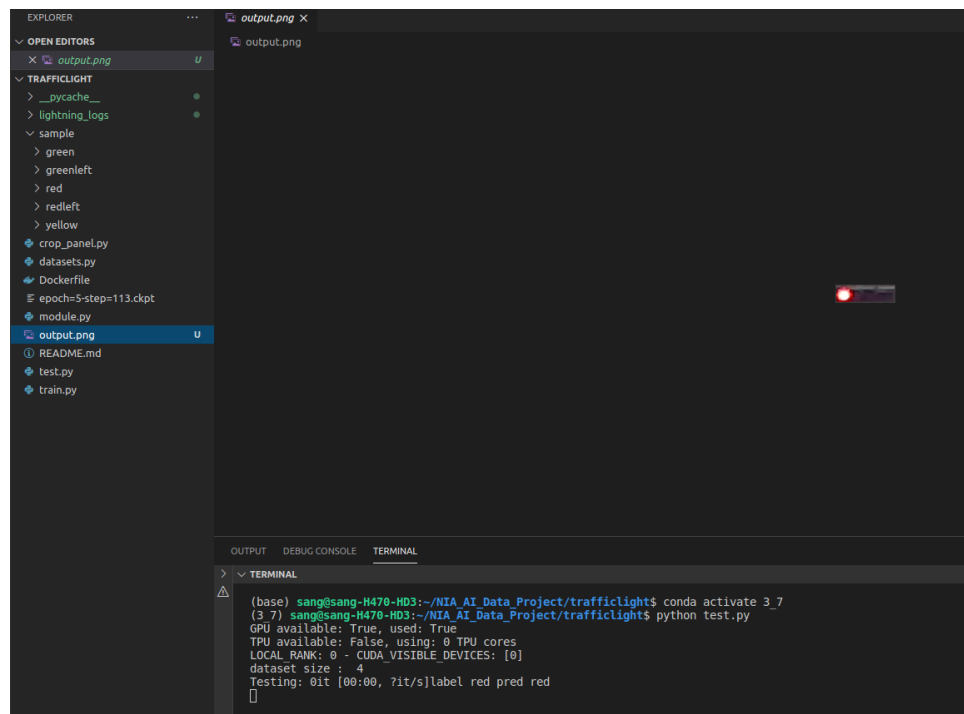
1. sample 데모 코드 실행 방법

1. 코드를 실행하기 위해서 필요한 파이썬 패키지를 설치합니다.

- numpy
- torch
- torchvision
- pytorch-lightning
- opencv-python
- matplotlib
- ujson

2. 패키지가 준비됐으면, python test.py 를 터미널에서 입력합니다.

3. python test.py로 테스트가 시작되면, 현재 테스트하고 있는 샘플 데이터가 아래와 같이 output.png라는 파일로 저장되고 터미널에 레이블값과 예측값을 출력합니다. 터미널에서 엔터를 누르면 그 다음 샘플 데이터를 읽으며 레이블값과 예측값을 출력합니다.



2. 대용량 데이터셋을 모델에서 사용하는 방법

1. 데이터셋 준비

1. 이미지 파일과 json파일로 이루어진 한 쌍의 데이터에 대해 신호등만 crop한 후 저장하는 작업이 필요합니다. 이부분은 crop_panel.py 참고하여 구현할 수 있습니다.

2. train.py 실행 방법

1. sample 폴더와 같은 폴더 구조로 train 폴더를 만들고 green, greenleft, red, redleft, yellow 레이블에 맞게 crop된 이미지를 저장하면 됩니다.
2. 데이터가 준비되었으면, python train.py를 실행하여 학습을 진행할 수 있습니다.
3. 학습이 진행되면 진행 상태를 나타내는 프로그레스 바와 현재의 epoch값이 표시됩니다.

```
| Name | Type | Params |
-----|-----|-----|
0 | fcn | MobileNetV2 | 2.2 M |
-----|-----|-----|
2.2 M | Trainable params |
0 | Non-trainable params |
2.2 M | Total params |
8.921 | Total estimated model params size (MB) |
dataset size : 4 |
/home/sang/anaconda3/envs/3.7/lib/python3.7/site-packages/pytorch_lightning/utilities/distributed.py:68: UserWarning: The dataloader, val dataloader 0, does not have many workers which may be a bottleneck. Consider increasing the value of the 'num_workers' argument (try 16 which is the number of cpus on this machine) in the 'DataLoader' init to improve performance.
warnings.warn(*args, **kwargs)
Validation sanity check: 56% | 1/2 [00:00<00:00, 2.66it/s]
dataset size : 4 |
/home/sang/anaconda3/envs/3.7/lib/python3.7/site-packages/pytorch_lightning/utilities/distributed.py:68: UserWarning: The dataloader, train dataloader, does not have many workers which may be a bottleneck. Consider increasing the value of the 'num_workers' argument (try 16 which is the number of cpus on this machine) in the 'DataLoader' init to improve performance.
warnings.warn(*args, **kwargs)
Epoch 14: 100% | 8/8 [00:00<00:00, 31.97it/s, loss=1.32, v_num=8]
(3.7) sangsang-H470-H03:~/NIA_AI_Data_Project/trafficlight$
```

1. test.py 실행 방법

1. 학습 때와 마찬가지로 sample 폴더와 같은 폴더 구조로 test 폴더를 만들고 학습시에 저장되었던 가중치 파일의 경로를 python test.py코드의 4번째 줄의 checkpoint_path의 경로에 입력해준 후, 테스트를 진행하면 됩니다.
2. python test.py를 실행하면 앞서 샘플 데이터를 테스트 했을때처럼 진행이 됩니다.