

A.I TEAM PROJECT

신윤규
권경환
이준모
김창우
한근희



LIST

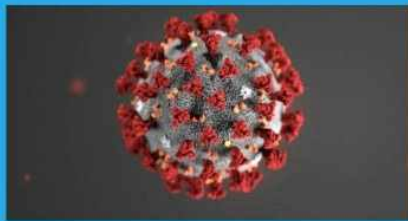
딥러닝 목적
데이터 저장과 전처리
딥러닝 모델
훈련과정
예측결과의 시각화



딥러닝 목적

Prezi

마스크를 쓴 사람을 구별하는 인공지능을 만들게 된 이유



Prezi

데이터 저장과 전처리

데이터 수집

- 1 마스크 착용을 구별하는 AI
- 2 마스크를 쓴 사진을 수집
- 3 종류를 4가지로 분류
- 4 각 종류별로 100장씩 수집

마스크 사진을 4 종류로 분류하여 저장

1

애니메이션



2

사람



3

동물



4

동물 마스크 쓴 사람



X100



딥러닝 모델



모델 생성을 위한 코드들

```
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))
```

```
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')
```

사용할 사진을 불러오는 코드

```
import yaml
with open('/content/data.yaml', 'r') as f:
    data = yaml.load(f)
print(data)
data['train'] = '/content/train.txt'
data['val'] = '/content/val.txt'
with open('/content/data.yaml', 'w') as f:
    yaml.dump(data, f)
print(data)
```



모델의 요약된 구조

Model Summary: 232 layers, 7249215 parameters, 0 gradients

LAYERS

232

PARAMETERS

7249215

GRADIENTS

0



훈련과정



모델 훈련을 위한 코드

```
#모델 훈련 코드  
%cd /content/yolov5/  
!python train.py --img 416 --batch 16 --epochs 100 --data /content/data.yaml --cfg /models/yolov5s.yaml --weights yolov5s.pt --name gun_yolov5s_results
```

훈련을 총 100회 진행

```
%load_ext tensorboard  
!tensorboard --logdir /content/yolov5/runs/
```

```
!python detect.py --source ./mask.jpg --weights ./runs/train/gun_yolov5s_results/weights/best.pt
```



첫 번째 훈련/백 번째 훈련

epoch	loss	box	obj	cls	total	targets	img_size
0x 0/5	100.00%	111/5	111/5	111/5	111/5	111/5	111/5
0/30	1.624	0.1244	0.08992	0.02783	0.2422	25	416
cls	loss	targets	targets	targets	targets	targets	targets
all	21	103	0	0	0.00145	0.00029	

box - 0.1244

obj - 0.08992

cls - 0.02783

total - 0.2422

box - 0.04798

obj - 0.08049

cls - 0.1396

total - 0.1424



예측결과의 시각화



훈련 시킨 모델을 테스트하는 코드

```
#학습시킨 모델을 부르기 위한 이미지들은 테스트해보는 코드  
#이미지들을 분석하여 마스크를 생성하는 코드는 주석으로 놓아두고 불러와 주석으로 해제하고 --source에 해당할터영상을 지정하면 실행이 됨.  
python detect.py --weights ./runs/train/gan_yolov5s_results/weights/test.pt --img 416 --conf 0.4 --source ./content/ask_animation
```

```
python detect.py --weights ./runs/train/gan_yolov5s_results/weights/test.pt --img 416 --conf 0.4 --source ./content/ask_animation
```

```
import glob  
from IPython.display import Image, display  
  
for imageName in glob.glob('./content/yolov5/runs/detect/exp11/*.*jpg'):  
    display(Image(filename=imageName))  
    print('👤')
```

학습시킨 모델로 테스트
를 진행하는 코드

테스트한 결과를 이미
지로 보여주는 코드

각 종류별 실행 결과

애니메이션



동물



동물마스크



사진



감사합니다

