

2023년 07월 05일

받은 파일 리스트

영상 파일

- 작동 영상 파일

이미지 파일

- 라벨을 제거한 페트병 이미지 파일
- 전력차단 이미지 파일
- 상자 테이프 제거 이미지 파일
- 찌그러진 캔 이미지 파일

모델 파일

- first_model3.h5 - 1차 분류 모델 → 5개로 분류한다. (캔, 상자, 페트, 전력, 그 외)
- model_f.h5 - 1차 분류 모델 → 위 모델과 동일 1차 분류 모델은(first_model.h5, model_f.h5) 둘 중 하나를 사용하면 됨
- can_model1.h5 - 캔, 찌그러진 캔, 그 외 분류 모델
- box_model1.h5 - 상자, 그 외 분류 모델 → 상자만 분류
- pet_model1.h5 - 페트병, 라벨 제거된 페트병, 그 외 분류 모델
- outlet_model1.h5 - 전력 그 외 분류 모델 → 전력이 사용 되는지 안사용 되는지 유무 판별 못함.
- 1차 분류모델: inception_resnet-v2
- 2차 분류모델: model: resnet50

코드 파일

mobilenetv3.box.ipynb - mobilenet을 사용한 박스 분류 모델 학습시키는 코드

mobilenetv3.can.ipynb - Can 분류 모델 학습시키는 코드

mobilenetv3.outlet.ipynb - 전력 분류 학습시키는 코드

mobilenetv3.pet.ipynb - 페트병 분류 학습시키는 코드

webcam.py - 모델 파일 불러와 웹캠으로 분류, 키입력 Q로 프로그램 종료

*jupyternotebook 이나 googlecolab에서는 서버에 연결되어있어 webcam.py 실행 못함

visual studio code에 모델파일과 webcam.py을 넣어 실행
페트병 인식할때 프로그램 다운

코드 수정

webcam.py

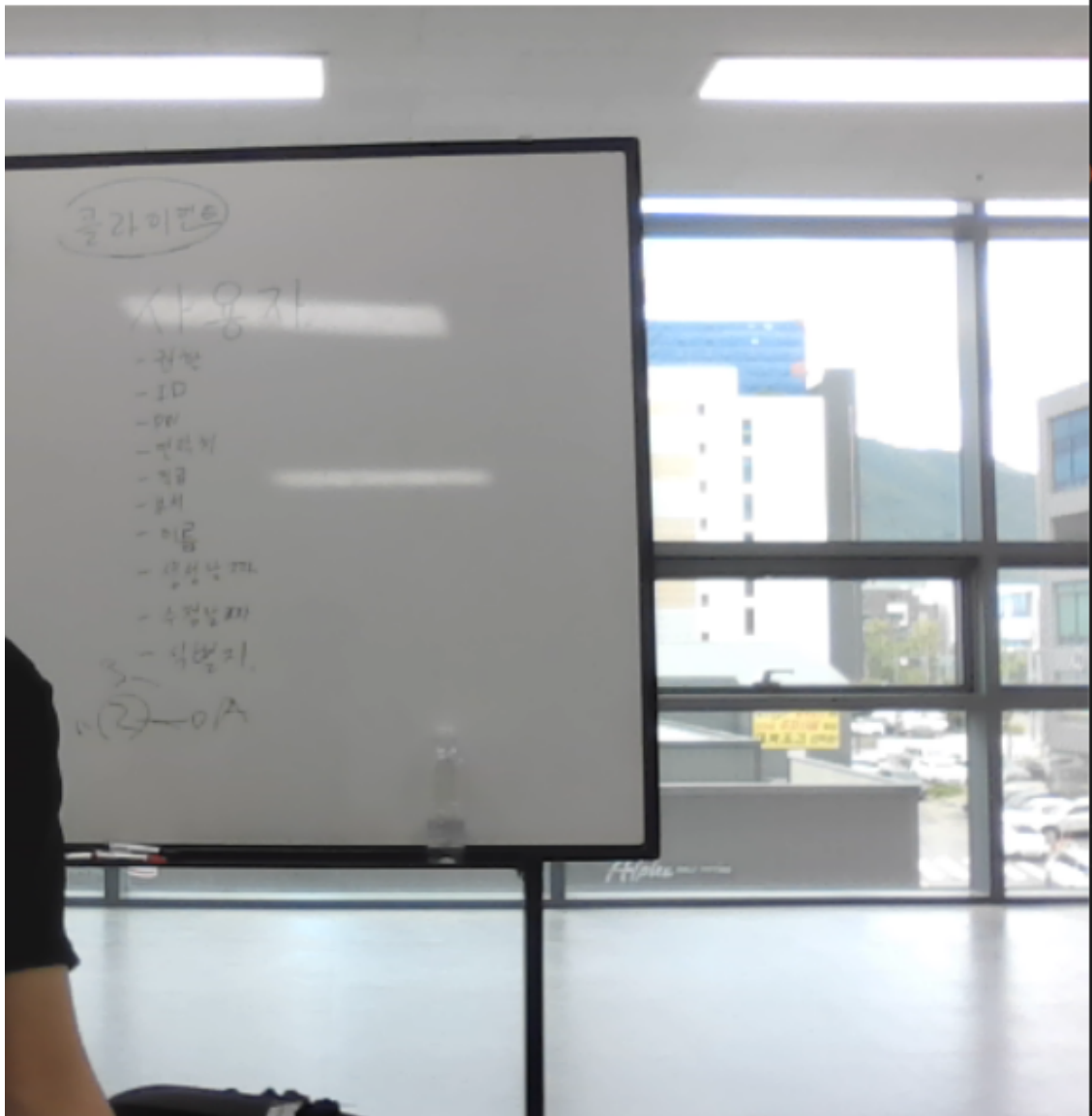
line 125

if pred[0]>0.7: → if np.max(pred)>0.7:

가까이 있는 물체는 잘 분류하지만 멀리 있는 물체는 분류를 못한다.

카메라 성능 문제 or 가까이 있는 물체만 학습 추정







모바일에서도 사용 가능하게 .h5 → .tflite로 변환

```
import tensorflow as tf

# Load the Keras model from the H5 file
keras_model = tf.keras.models.load_model('/content/drive/MyDrive/data/model/pet_model1.h5')

# Convert the Keras model to a TFLite model
converter = tf.lite.TFLiteConverter.from_keras_model(keras_model)
tflite_model = converter.convert()

# Save the TFLite model to a file
with open('/content/drive/MyDrive/data/model/pet_model1.tflite', 'wb') as f:
    f.write(tflite_model)
```

 pet_model1.h5
 pet_model1.tflite