

인공지능 발표

Let's Safe!

Yongwon KIM

Presentation

Case Similarity: 0.0

PASS



안전하세요!

산업현장 안전을 위한
안전모 탐지 시스템 구현

제주대학교 컴퓨터공학전공

2022208020
김용원

인공지능 발표

CONTENTS

Yongwon KIM

ORIENTATION

MIRI COMPANY

목차

CONTENTS

1. 개요, 필요성
2. 관련 연구/내용
3. 내용 요약
4. 캐글
5. 코드 설명
6. 결론
7. 질의응답



개요/필요성

INTRODUCTION



TF

Keras

openCV

개요

AI 기술을 이용한 안전모 감지 서비스 진행하여
새로운 시대에 맞춘 진보한 시스템 구축,
더 정확하고 더 빠른 서비스 제공

이제는 안전입니다!

개요/필요성

INTRODUCTION

국민일보

건설현장 33% 안전모도 안 썼다... 지난해 236명 추락 사망

[벼랑 끝 노동자 안전] ③ 추락위험 일제점검 <끝>

입력 2021-07-20 04:04



사진=게티이미지

전국 건설현장 3곳 중 1곳은 안전모 미착용 등 개인 보호구를 제대로 갖추지 않고 있다는 정부 조사 결과가 나왔다. 안전난간·작업 발판을 설치하지 않은 공사장도 상당수였다. 노동자 생명을 보호하는 최소한의 안전 체계마저 정상 작동하지 않고 있다는 지적이다. 이에 고용노동부와 한국산업안전보건공단은 이달부터 산업재해 현장점검을 정례화해 ‘안전한 일터’ 조성에 본격 나서기로 했다.

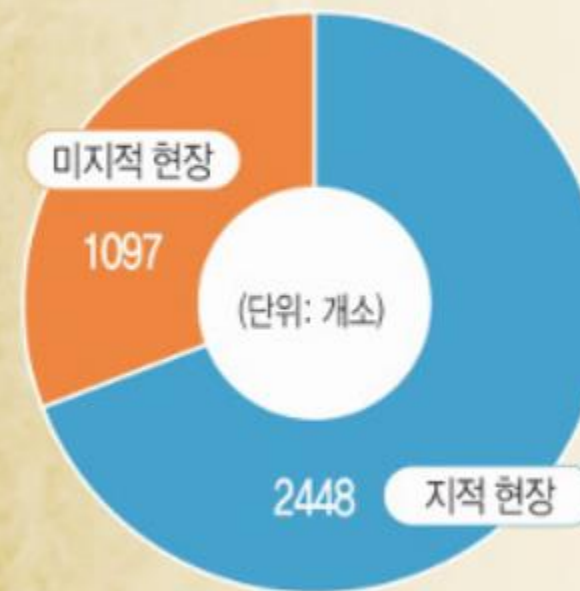
Why

So

Safe

필요성

전국 건설현장 추락위험 점검결과



점검내용별 지적현장 (단위: 개소)



2020년 산업재해 사망자 882명 중
건설업 사망자 수 (51.9%) -고용노동부-



458명

관련 연구/내용

RESEARCH

관련 현황

LG

U+

"안전모 미착용자 실시간 감지"...스마트팩토리 사업 뛰어든 LGU+

윤정민 기자 | 입력 2023. 3. 12. 09:00

0

🔍 🔊 🔒 🔗 🖨

기사내용 요약

LGU+, 2023 스마트공장·자동화산업전서 '스마트팩토리' 선보여
중대재해처벌법 시행 후 지능형 보안 솔루션 문의 250% 늘어



[서울=뉴스시스] 윤정민 기자 = 지난 9일 서울 코엑스에서 열린 '2023 스마트공장·자동화산업전'에서 LG유플러스가 선보인 디지털트윈 기반 통합관제센터 솔루션. 한 근로자가 안전모를 착용하지 않은 모습이 폐쇄회로(CC)TV에 찍히자 통합관제센터에 알림이 울리고 있다. 2023.03.09. alpaca@newsis.com *재판매 및 DB 금지

SK

SHILDERS

SK실더스, SK텔레콤-SK플래닛과 ICT기반 산업안전장치 출시

발행일: 2022-03-08 10:36 | 지면: 2022-03-09 | 12면



SK 실더스

SK실더스가 영상통합관제기술과 SK텔레콤 무선통신기술-SK플래닛 센싱기술을 융합한 정보통신기술(ICT)기반 이동형 산업안전장치를 출시했다.

관련 연구/내용

RESEARCH

관련 현황

HYUNDAI

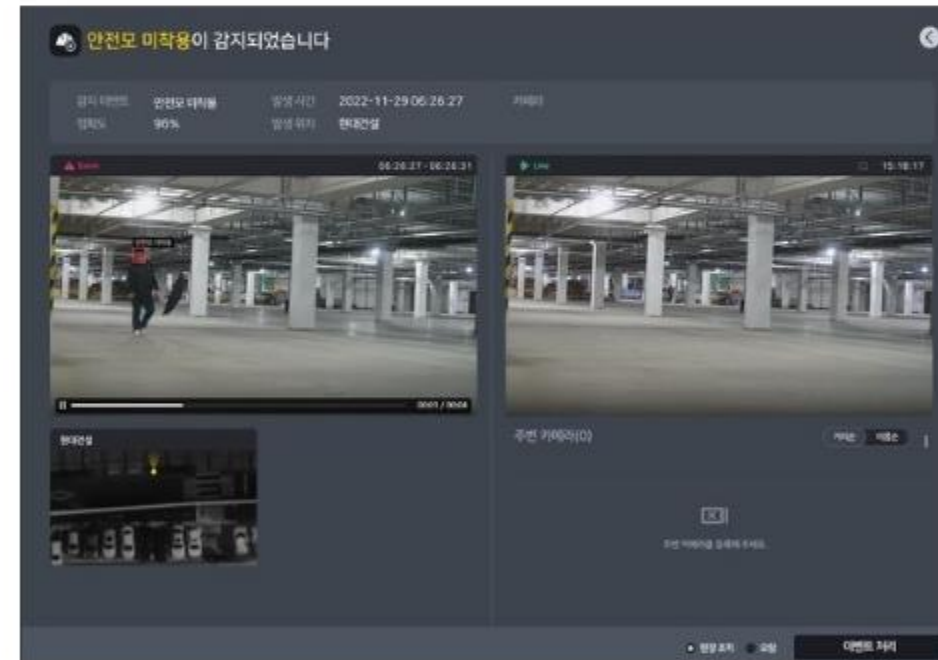
ELEVATOR

'안전모 미착용자 감지' 알람..AI로 현장 안전 관리

한민형 기자 | 입력 2022.12.05 15:47 | 댓글 0



현대건설, 현장 CCTV 영상 분석 시스템 자체 개발
AI 영상 분석 활용해 건설 현장 안전·품질 관리
레미콘 품질 실시간 모니터링 시스템도 개발



* 현대건설이 현장 CCTV 영상 분석 시스템 안전 위반사항 탐지 및 알람을 제공한다

AI(인공지능)기술이 건설 현장의 안전관리에 도입, 현장의 안전사고가 감소할 것으로 보인다. 현대건설은 AI(인공지능) 기반의 건설 현장 영상 분석 기술을 개발하고 특허를 등록하는 등 스마트 건설 관리에 속도를 내고 있다고 5일 밝혔다.

SAMSUNG

S-1

경제 : 경제일반

안전모 벗자 앵~ 경고음...사고 줄이는 똑똑한 CCTV

중앙일보 | 입력 2023.08.04 00:01

자면보기

박해리 기자 구독



지난 1일 서울 에스원 R&D센터에서 본지 기자가 SVMS를 체험하고 있다. 안전모를 벗으니 CCTV가 이를 인식해 경고음이 울렸다. [사진 에스원]

서울 중구 순화동 에스원 사옥에 있는 연구개발(R&D)센터. 지난 1일 찾은 이곳에서는 지능형 폐쇄회로(CC)TV를 활용해 공장 내 작업자들의 움직임을 분석하는 실험이 진행되고 있었다.

기자는 이날 노란색 작업복과 안전모를 착용하고 안전수칙 테스트에 참여했다. 카메라 앞에 서자 가로 2.5m, 세로 1.5m짜리 대형 화면의 16개 모니터에 기자의 모습이 나타났다. 안전모를 벗자 불과 5초도 안 돼 '앵~' 하고 소란스러운 경고음이 울리고, 화면에는 빨간색 알람 표시가 떴다. 이 알람은 실시간으로 관리자의 스마트폰 애플리케이션(앱)으로 전송됐다.

요약

SUMMARY

안전모 착용 중요

목숨은 소중하니까요



대기업이 진행 중

중대재해처벌법 시행 이후 대기업 도입 늘어
삼성, LG, 현대건설 등



It's Possible

대부분의 대기업이 홍보하는 시스템 한계 존재
장비 포함 모든 시스템 판매

별도의 카메라 장비 구매가 필요없다면?
It's Possible



그게
돈이 됩니다-

캐글



JOSU! 너로 정했다!

캐글

왓지 모르게 신뢰가 가는 프로필 사진

5500여개의 높은 조회수

155개의 Copy & Edit



JOSU · 3Y AGO · 5,561 VIEWS

▲

15

Copy & Edit

155

Tensorflow 2 Safety Helmet Detector

Python · [Safety Helmet Detection](#)

Notebook

Input

Output

Logs

Comments (0)

Run

3.9s

Version 1 of 2

TensorFlow

BBox helmet detector with Tensorflow (>=) 2.3 and Retinanet (SSD with Resnet 50 v1) Object detection model on COCO 2017 dataset

- Suggested Metrics in last cell

In [1]:

```
import os

TF_FOLDER = 'TF23_SSD_RESNET50'
WORKSPACE = 'helmet'
PROJECT = 'detector'

# Bounding Box classes
CLASSES = ['helmet', 'head', 'person']

RUN_DIR = os.getcwd() # Current dir
TF_DIR = RUN_DIR + "/" + TF_FOLDER
TF_PROJECT_DIR = TF_DIR + "/" + WORKSPACE + "/" + PROJECT
MODELS_DIR = RUN_DIR + "/" + TF_FOLDER + "/models"
RESEARCH_DIR = MODELS_DIR + "/research"

# Pre-Trained model from Tensorflow 2 model zoo
PRE_MODEL_FILE = 'ssd_resnet50_v1_fpn_640x640_coco17_tpu-8.tar.gz'
PRE_MODEL_URL = 'http://download.tensorflow.org/models/object_detection/tf2/20200711/ssd_resnet50_v1_fpn_640x640_coco17_tpu-8.tar.gz'
PRE_MODEL_TYPE = 'ssd_resnet50_v1_fpn_640x640_coco17_tpu-8'
PRE_MODEL_CHECKPOINT = PRE_MODEL_TYPE + '/checkpoint/ckpt-0'
MY_PRE_MODEL_DIR = TF_PROJECT_DIR + '/models/ssd_resnet50_v1_fpn'

# Kaggle dataset paths
DATA_XML_DIR = '/kaggle/input/hard-hat-detection/annotations/'
```


캐글

Kaggle

그러나 이 코드가 사용되는 일은 없었다

안되는데요?

```
notebook0e649fdead Draft saved
File Edit View Run Help

Explore and run machine learning code with Kaggle Notebooks. Sign in or Register to continue editing.

+ ✂ 📄 📌 ▶ ⏮ Run All Code ▾

RESEARCH_DIR = MODELS_DIR + "/research"

# Pre-Trained model from TensorFlow 2 model zoo
PRE_MODEL_FILE = "ssd_resnet50_v1_fpn_640x640_coco17_tpu-8.tar.gz"
PRE_MODEL_URL = "http://download.tensorflow.org/models/object_detection/tf2/20200711/ssd_resnet50_v1_fpn_640x640_coco17_tpu-8.tar.gz"
PRE_MODEL_TYPE = "ssd_resnet50_v1_fpn_640x640_coco17_tpu-8"
PRE_MODEL_CHECKPOINT = PRE_MODEL_TYPE + "/checkpoint/ckpt-0"
MY_PRE_MODEL_DIR = TF_PROJECT_DIR + "/models/ssd_resnet50_v1_fpn"

# Kaggle dataset paths
DATA_XML_DIR = "/kaggle/input/hard-hat-detection/annotations/"
DATA_IMAGES_DIR = "/kaggle/input/hard-hat-detection/images/"

STEPS = 10000
CHECKPOINT_EVERY = 1000
BATCH_SIZE = 4

# Create project folder and install TensorFlow 2.3
import pathlib
import sys
from os import path
os.environ['TF_CPP_MIN_LOG_LEVEL'] = '2'

!mkdir -p $TF_FOLDER
!cd $TF_FOLDER
!{sys.executable} -m pip install --upgrade tensorflow

/kaggle/working/TF23_SSD_RESNET50
WARNING: Retrying (Retry(total=4, connect=None, read=None, redirect=None, status=None)) after connection broken by 'NewConnectionError(<pip_vendor.urllib3.connection.HTTPSConnection object at 8x7d9e234e38d0>: Failed to establish a new connection: [Errno -3] Temporary failure in name resolution)': /simple/tensorflow/
WARNING: Retrying (Retry(total=3, connect=None, read=None, redirect=None, status=None)) after connection broken by 'NewConnectionError(<pip_vendor.urllib3.connection.HTTPSConnection object at 8x7d9e2348b35b>: Failed to establish a new connection: [Errno -3] Temporary failure in name resolution)': /simple/tensorflow/
WARNING: Retrying (Retry(total=2, connect=None, read=None, redirect=None, status=None)) after connection broken by 'NewConnectionError(<pip_vendor.urllib3.connection.HTTPSConnection object at 8x7d9e2348b19b>: Failed to establish a new connection: [Errno -3] Temporary failure in name resolution)': /simple/tensorflow/
WARNING: Retrying (Retry(total=1, connect=None, read=None, redirect=None, status=None)) after connection broken by 'NewConnectionError(<pip_vendor.urllib3.connection.HTTPSConnection object at 8x7d9e2348b59b>: Failed to establish a new connection: [Errno -3] Temporary failure in name resolution)': /simple/tensorflow/
WARNING: Retrying (Retry(total=0, connect=None, read=None, redirect=None, status=None)) after connection broken by 'NewConnectionError(<pip_vendor.urllib3.connection.HTTPSConnection object at 8x7d9e234893db>: Failed to establish a new connection: [Errno -3] Temporary failure in name resolution)': /simple/tensorflow/
Requirement already up-to-date: tensorflow in /opt/conda/lib/python3.7/site-packages (2.3.0)
Requirement already satisfied, skipping upgrade: h5py<2.11.0,>=2.10.0 in /opt/conda/lib/python3.7/site-packages (from tensorflow) (2.10.0)
Requirement already satisfied, skipping upgrade: wheel<0.36 in /opt/conda/lib/python3.7/site-packages (from tensorflow) (0.34.2)
Requirement already satisfied, skipping upgrade: google-pasta<0.1.5 in /opt/conda/lib/python3.7/site-packages (from tensorflow) (0.2.0)
Requirement already satisfied, skipping upgrade: wrapt<1.11.1 in /opt/conda/lib/python3.7/site-packages (from tensorflow) (1.11.2)
Requirement already satisfied, skipping upgrade: astunparse<=1.6.3 in /opt/conda/lib/python3.7/site-packages (from tensorflow) (1.4.3)
WARNING: Retrying (Retry(total=4, connect=None, read=None, redirect=None, status=None)) after connection broken by 'NewConnectionError(<pip_vendor.urllib3.connection.HTTPSConnection object at 8x7d9e234528d0>: Failed to establish a new connection: [Errno -3] Temporary failure in name resolution)': /simple/tensorboard/
WARNING: Retrying (Retry(total=3, connect=None, read=None, redirect=None, status=None)) after connection broken by 'NewConnectionError(<pip_vendor.urllib3.connection.HTTPSConnection object at 8x7d9e2345271b>: Failed to establish a new connection: [Errno -3] Temporary failure in name resolution)': /simple/tensorboard/
WARNING: Retrying (Retry(total=2, connect=None, read=None, redirect=None, status=None)) after connection broken by 'NewConnectionError(<pip_vendor.urllib3.connection.HTTPSConnection object at 8x7d9e234527db>: Failed to establish a new connection: [Errno -3] Temporary failure in name resolution)': /simple/tensorboard/
WARNING: Retrying (Retry(total=1, connect=None, read=None, redirect=None, status=None)) after connection broken by 'NewConnectionError(<pip_vendor.urllib3.connection.HTTPSConnection object at 8x7d9e2345281b>: Failed to establish a new connection: [Errno -3] Temporary failure in name resolution)': /simple/tensorboard/
WARNING: Retrying (Retry(total=0, connect=None, read=None, redirect=None, status=None)) after connection broken by 'NewConnectionError(<pip_vendor.urllib3.connection.HTTPSConnection object at 8x7d9e234529db>: Failed to establish a new connection: [Errno -3] Temporary failure in name resolution)': /simple/tensorboard/
ERROR: Could not find a version that satisfies the requirement tensorflow<3, >=2.3.0 (from tensorflow) (from versions: none)
ERROR: No matching distribution found for tensorflow<3, >=2.3.0 (from tensorflow)
```

```
Downloading data from https://github.com/tensorflow/models/archive/master.zip
-----
galerror: Traceback (most recent call last)
/opt/conda/lib/python3.7/urllib/request.py in do_open(self, http_class, req, **http_conn_args)
1318     h.request(req.get_method(), req.selector, req.data, headers,
-> 1319               encode_chunked=req.has_header('Transfer-encoding'))
1320     except OSError as err: # timeout error

/opt/conda/lib/python3.7/http/client.py in request(self, method, url, body, headers, encode_chunked)
1251     """Send a complete request to the server."""
-> 1252     self._send_request(method, url, body, headers, encode_chunked)
1253

/opt/conda/lib/python3.7/http/client.py in _send_request(self, method, url, body, headers, encode_chunked)
1297     body = _encode(body, 'body')
-> 1298     self._endheaders(body, encode_chunked=encode_chunked)
1299

/opt/conda/lib/python3.7/http/client.py in _endheaders(self, message_body, encode_chunked)
1246     raise CannotSendHeader()
-> 1247     self._send_output(message_body, encode_chunked=encode_chunked)
1248

/opt/conda/lib/python3.7/http/client.py in _send_output(self, message_body, encode_chunked)
1025     del self._buffer[:]
-> 1026     self.send(msg)
1027

/opt/conda/lib/python3.7/http/client.py in send(self, data)
965     if self.auto_open:
-> 966         self.connect()
967     else:

/opt/conda/lib/python3.7/http/client.py in connect(self)
1413
-> 1414     super().connect()
1415

/opt/conda/lib/python3.7/http/client.py in connect(self)
937     self.sock = self._create_connection(
-> 938         (self.host, self.port), self.timeout, self.source_address)
939     self.sock.setsockopt(socket.IPPROTO_TCP, socket.TCP_NODELAY, 1)

/opt/conda/lib/python3.7/socket.py in create_connection(address, timeout, source_address)
706     err = None
-> 707     for res in getaddrinfo(host, port, 0, SOCK_STREAM):
708         af, socktype, proto, canonname, sa = res

/opt/conda/lib/python3.7/socket.py in getaddrinfo(host, port, family, type, proto, flags)
751     addrlist = []
-> 752     for res in _socket.getaddrinfo(host, port, family, type, proto, flags):
753         af, socktype, proto, canonname, sa = res

galerror: [Errno -3] Temporary failure in name resolution

During handling of the above exception, another exception occurred:

URLError: Traceback (most recent call last)
/opt/conda/lib/python3.7/site-packages/tensorflow/python/keras/utils/data_utils.py in get_file(fname, origin, untar, md5_hash, file_hash, cache_subdir, hash_algorithm, extract, archive_format, cache_dir)
277     try:
-> 278         urlretrieve(origin, fpath, dl_progress)
279     except HTTPError as e:

/opt/conda/lib/python3.7/urllib/request.py in urlretrieve(url, filename, reporthook, data)
246
-> 247     with contextlib.closing(urlopen(url, data)) as fp:
248         headers = fp.info()

/opt/conda/lib/python3.7/urllib/request.py in urlopen(url, data, timeout, cafile, capath, cadefault, context)
221     opener = _opener
-> 222     return opener.open(url, data, timeout)
223

/opt/conda/lib/python3.7/urllib/request.py in open(self, fullurl, data, timeout)
524
-> 525     response = self._open(req, data)
526
```

대안발표

Back-up

이제부터 내가 직접 나서겠다

대안 발표



발표

It's Mine

간단 설명

트레이너



실행기



STEP 2

Visualizer

```
test.py train_mask_detector.py
1 from tensorflow.keras.applications.mobilenet_v2 import preprocess_input
2 from tensorflow.keras.preprocessing.image import img_to_array
3 from tensorflow.keras.models import load_model
4 from imutils.video import VideoStream
5 import numpy as np
6 import imutils
7 import time
8 import cv2
9 import matplotlib.pyplot as plt
10
11 # 외부 카메라와 연결시 안보관으로 인한 지연현상 있어서 강제루 값을 넣어들
12 prev_time = 0
13 FPS = 30
14
15 # Case Similarity를 저장할 리스트
16 safetyhelmet_similarities = []
17
18 def detect_and_predict_hat(frame, faceNet, hatNet):
19     # 프레임의 차수 잡기
20     (h, w) = frame.shape[:2]
21     blob = cv2.dnn.blobFromImage(frame, 1.0, (224, 224), (104.0, 177.0, 123.0))
22
23     # 얼굴 감지
24     faceNet.setInput(blob)
25     detections = faceNet.forward()
26
27     # 얼굴 복원, 해당 위치 및 얼굴 마스크 네트워크의 예측 복원을 추가화
28     faces = []
29     locs = []
30     preds = []
31
32     # 할지 img
33     for i in range(0, detections.shape[2]):
34         # 확률 추출
35         confidence = detections[0, 0, i, 3]
36         # 작은 확률 버리기
37         if confidence > 0.5:
38             # 객체에 대한 관계 상자의 (x, y) 좌표를 계산
39             box = detections[0, 0, i, 3:7] * np.array([w, h, w, h])
40             (startX, startY, endX, endY) = box.astype("int")
41
42             # 박스 크기
43             (startX, startY) = (max(0, startX), max(0, startY))
44             (endX, endY) = (min(w - 1, endX), min(h - 1, endY))
45
46             # 카메라
47             face = frame[startY:endY, startX:endX]
48             face = cv2.cvtColor(face, cv2.COLOR_BGR2RGB)
49             face = cv2.resize(face, (224, 224))
50             face = img_to_array(face)
51             face = preprocess_input(face)
52
53             faces.append(face)
54             locs.append((startX, startY, endX, endY))
55
56 # 얼굴만 나열함의 예측
```

STEP 1

Trainer

```
1 Con_detector train_mask_detector.py
2 from tensorflow.keras.preprocessing.image import ImageDataGenerator
3 from tensorflow.keras.applications import MobileNetV2
4 from tensorflow.keras.layers import AveragePooling2D
5 from tensorflow.keras.layers import Dropout
6 from tensorflow.keras.layers import Flatten
7 from tensorflow.keras.layers import Dense
8 from tensorflow.keras.layers import Input
9 from tensorflow.keras.models import Model
10 from tensorflow.keras.optimizers import Adam
11 from tensorflow.keras.applications.mobilenet_v2 import preprocess_input
12 from tensorflow.keras.preprocessing.image import img_to_array
13 from tensorflow.keras.preprocessing.image import load_img
14 from tensorflow.keras.utils import to_categorical
15 from sklearn.preprocessing import LabelBinarizer
16 from sklearn.model_selection import train_test_split
17 from sklearn.metrics import classification_report
18 import matplotlib.pyplot as plt
19 import numpy as np
20 import os
21
22 # 배치 사이즈와 learning rate
23 INIT_LR = 1e-4
24 EPOCHS = 40
25 BS = 32
26
27 DIRECTORY = "C:/Users/kyatr/Desktop/1/Con_detector/model"
28 CATEGORIES = ["bald", "Safetyhelmet"]
29
30 # 이미지 가져오기
31 # (i.e., images)
32 print("[INFO] loading images...")
33
34 data = []
35 labels = []
36
37 for category in CATEGORIES:
38     path = os.path.join(DIRECTORY, category)
39     for img in os.listdir(path):
40         img_path = os.path.join(path, img)
41         image = load_img(img_path, target_size=(224, 224))
42         image = img_to_array(image)
43         image = preprocess_input(image)
44
45         data.append(image)
46         labels.append(category)
47
48 # 이분화 유틸리티
49 lb = LabelBinarizer()
50 labels = lb.fit_transform(labels)
51 labels = to_categorical(labels)
52
53 data = np.array(data, dtype="float32")
54 labels = np.array(labels)
```


발표

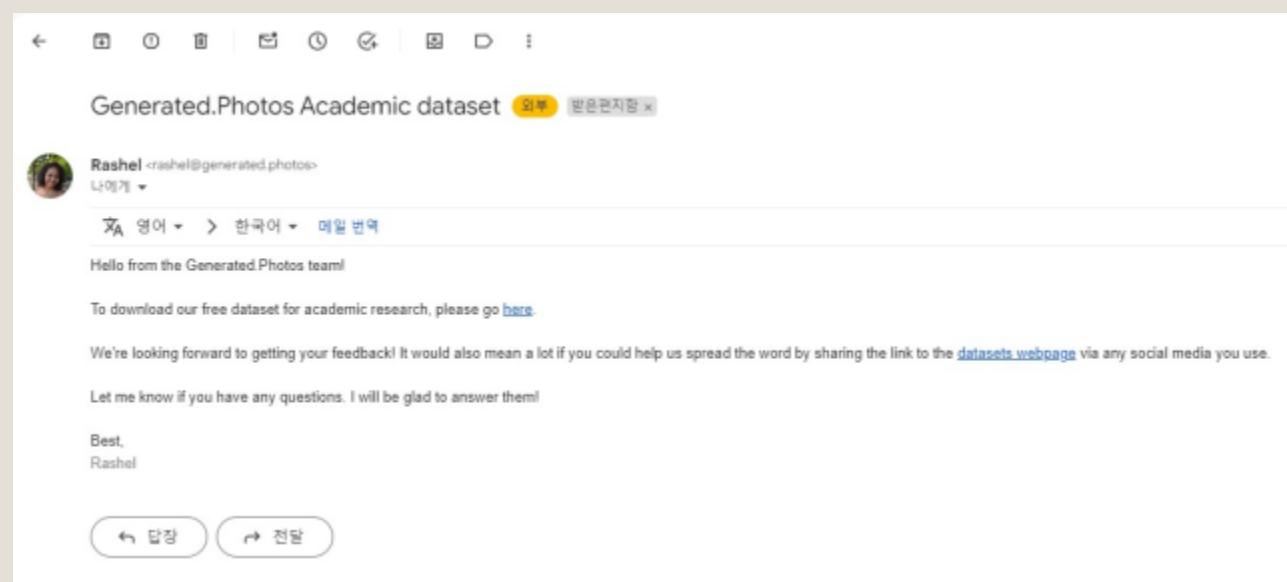
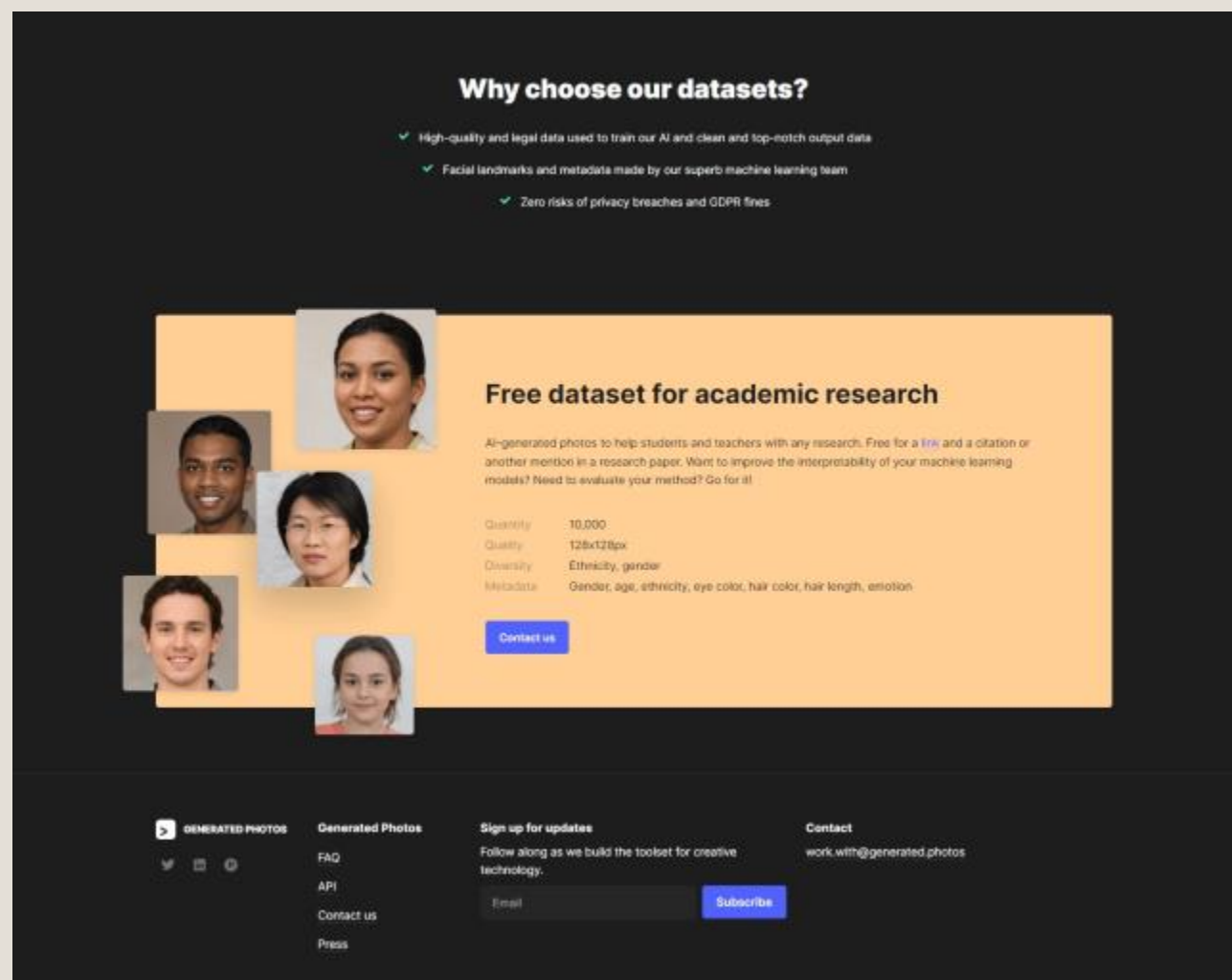
데이터?

More Dataaaaaaa!

트레이너



실행기



발표

트레이너 설명

Trainer

트레이너



실행기



```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.applications import MobileNetV2
from tensorflow.keras.layers import AveragePooling2D
from tensorflow.keras.layers import Dropout
from tensorflow.keras.layers import Flatten
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Input
from tensorflow.keras.models import Model
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.applications.mobilenet_v2 import preprocess_input
from tensorflow.keras.preprocessing.image import img_to_array
from tensorflow.keras.preprocessing.image import load_img
from tensorflow.keras.utils import to_categorical
from sklearn.preprocessing import LabelBinarizer
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
import matplotlib.pyplot as plt
import numpy as np
import os

# 배치 사이즈와 learning rate
INIT_LR = 1e-4
EPOCHS = 40
BS = 64

DIRECTORY = "C:/Users/kywtr/Desktop/1/Con_detector/model"
CATEGORIES = ["bald", "Safetyhelmet"]

# 이미지 가져오기
# (i.e., images)
print("[INFO] loading images...")

data = []
labels = []

for category in CATEGORIES:
    path = os.path.join(DIRECTORY, category)
    for img in os.listdir(path):
        img_path = os.path.join(path, img)
        image = load_img(img_path, target_size=(224, 224))
        image = img_to_array(image)
        image = preprocess_input(image)

        data.append(image)
        labels.append(category)
```

```
# 라벨에 원핫인코딩
lb = LabelBinarizer()
labels = lb.fit_transform(labels)
labels = to_categorical(labels)
data = np.array(data, dtype="float32")
labels = np.array(labels)

(trainX, testX, trainY, testY) = train_test_split(data, labels, test_size=0.20, stratify=labels, random_state=42)
# CUDA 버그 예외
# 데이터 증식을 위한 train img
aug = ImageDataGenerator(
    rotation_range=20,
    zoom_range=0.15,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.15,
    horizontal_flip=True,
    fill_mode="nearest")

# MobileNetV2 network 를 커서 fc layer가 켜져있는지 확인 / Google이 만든 DNN 아키텍처
baseModel = MobileNetV2(weights="imagenet", include_top=False, input_tensor=Input(shape=(224, 224, 3)))

# 기존 모델에 배치할 모델
headModel = baseModel.output
headModel = AveragePooling2D(pool_size=(7, 7))(headModel)
headModel = Flatten(name="flatten")(headModel)
headModel = Dense(128, activation="relu")(headModel)
headModel = Dropout(0.5)(headModel)
headModel = Dense(2, activation="softmax")(headModel)
model = Model(inputs=baseModel.input, outputs=headModel)

for layer in baseModel.layers:
    layer.trainable = False

# compile
print("[INFO] compiling model...")
opt = Adam(lr=INIT_LR, decay=INIT_LR / EPOCHS)
model.compile(loss="binary_crossentropy", optimizer=opt, metrics=["accuracy"])

# train
print("[INFO] training head...")
H = model.fit(
    aug.flow(trainX, trainY, batch_size=BS),
    steps_per_epoch=len(trainX) // BS,
    validation_data=(testX, testY),
    validation_steps=len(testX) // BS,
    epochs=EPOCHS)

# test set 예측
print("[INFO] evaluating network...")
predIdxs = model.predict(testX, batch_size=BS)

# 테스트 세트의 각 이미지에 대해 가장 큰 예측 확률을 가진 레이블의 인덱스 찾기
predIdxs = np.argmax(predIdxs, axis=1)

print(classification_report(testY.argmax(axis=1), predIdxs, target_names=lb.classes_))
```


발표

트레이너 설명

```
# 모델 저장
print("[INFO] saving mask detector model...")
model.save("mask_detector.model", save_format="h5")

# 손실률과 정확도 띄우기
N = EPOCHS

plt.style.use("ggplot")
plt.figure()

plt.plot(np.arange(0, N), H.history["loss"], label="train_loss")
plt.plot(np.arange(0, N), H.history["val_loss"], label="val_loss")
plt.plot(np.arange(0, N), H.history["accuracy"], label="train_acc")
plt.plot(np.arange(0, N), H.history["val_accuracy"], label="val_acc")

plt.title("Training loss and Accuracy")
plt.xlabel("Epoch #")
plt.ylabel("Loss/Accuracy")
plt.legend(loc="lower left")
plt.savefig("plot.png")
```

```

Run: train_mask_detector
2023-11-12 23:33:41.193312: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1510] Created device /job:localhost/replica:0/task:0/device:GPU:0 with 7435 MB memory: -- device: 0, name: NVIDIA GeForce RTX 3080, pci bus id: 0000:08:00.0, compute capability: 8.6
[INFO] compiling model...
C:\Users\kyle\Tr\code\data\local\Programs\Python\Python39\lib\site-packages\tensorflow\python\keras\tf_optimizer_v2\tf_optimizer_v2.py:376: UserWarning: The 'lr' argument is deprecated, use 'learning_rate' instead.
  warnings.warn(
[INFO] training head...
2023-11-12 23:33:42.702107: I tensorflow/compiler/mlir/mlir_graph_optimization_pass.cc:185] None of the MLIR Optimization Passes are enabled (registered 2)
Epoch 1/40
2023-11-12 23:33:44.888274: I tensorflow/stream_executor/cuda/cuda_dnn.cc:349] Loaded cuDNN version 8108
2023-11-12 23:33:45.617022: W tensorflow/stream_executor/gpu/rdzone/allocator.cc:314] Internal: ptxas exited with non-zero error code -1, output:
Relying on driver to perform ptx compilation.
Modify $PATH to customize ptxas location.
This message will be only logged once.
2023-11-12 23:33:46.564945: I tensorflow/stream_executor/cuda/cuda_blas.cc:1700] TensorFloat-32 will be used for the matrix multiplication. This will only be logged once.
131/131 [=====] - ETA: 0s - loss: 0.1003 - accuracy: 0.97052023-11-12 23:34:27.343188: W tensorflow/core/framework/cpu_allocator_impl.cc:80] Allocation of 1269252096 exceeds 10% of free system memory.
131/131 [=====] - 47s 333ms/step - loss: 0.1063 - accuracy: 0.9705 - val_loss: 0.0170 - val_accuracy: 0.9953
Epoch 2/40
90/131 [=====] - ETA: 12s - loss: 0.0249 - accuracy: 0.9943

```

[illegible]

트레이너

A square QR code with a green border, located below the '트레이너' (Trainer) text.

실행기

A square QR code with a red border, located in the bottom right section of the slide.

발표

트레이너 설명

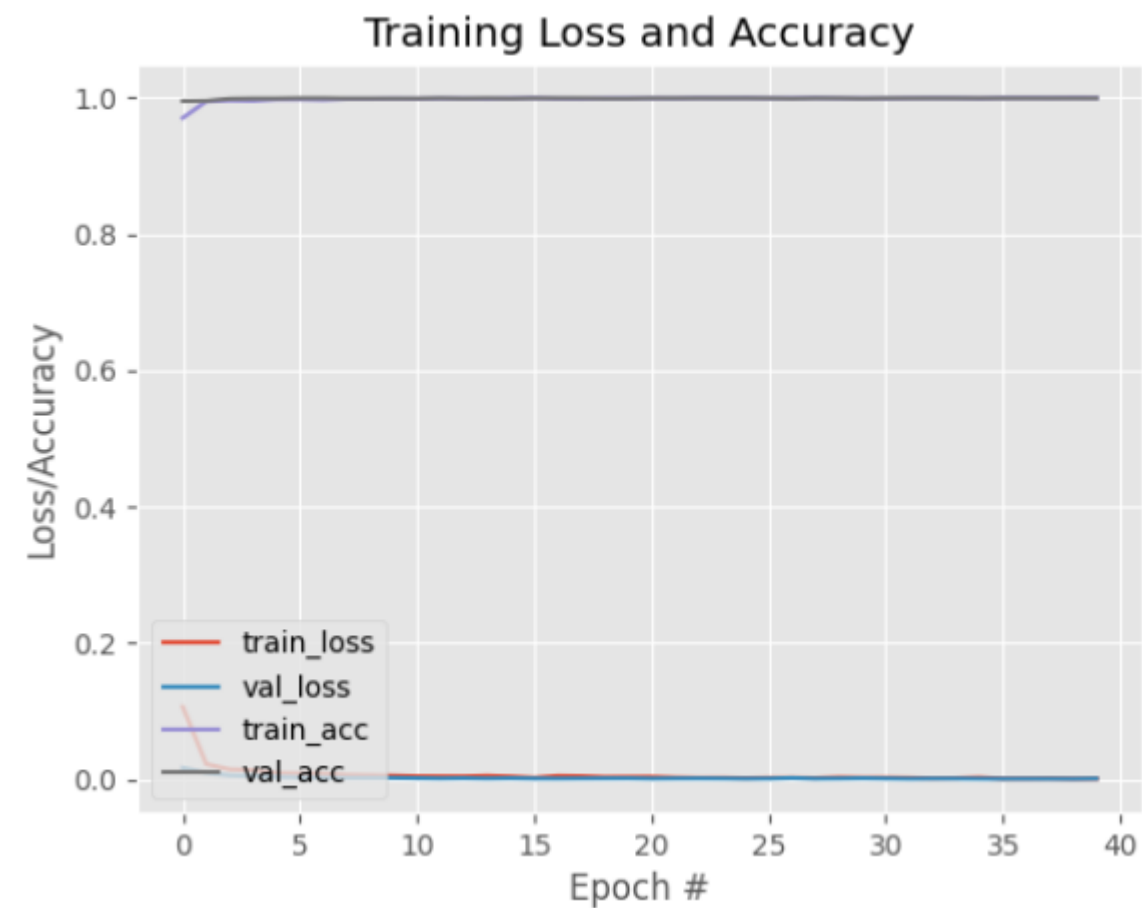
이름	수정된 날짜	유형	크기
dataset	2023-03-21 오후 4:33	파일 폴더	
face_detector	2023-03-22 오후 1:58	파일 폴더	
model	2023-06-13 오후 10:08	파일 폴더	
module	2023-06-13 오후 10:15	파일 폴더	
ddd	2023-03-21 오후 5:09	JetBrains PyChar...	1KB
mask_detector.model	2023-11-13 오전 12:01	MODEL 파일	11,230KB
plot	2023-11-13 오전 12:01	PNG 파일	27KB
sim	2023-11-13 오전 1:05	PNG 파일	53KB
similarity	2023-11-13 오전 1:05	텍스트 문서	3KB
test	2023-11-13 오전 1:04	JetBrains PyChar...	5KB
train_mask_detector	2023-11-14 오전 2:33	JetBrains PyChar...	4KB

Trainer

트레이너



실행기



발표

실행기 설명

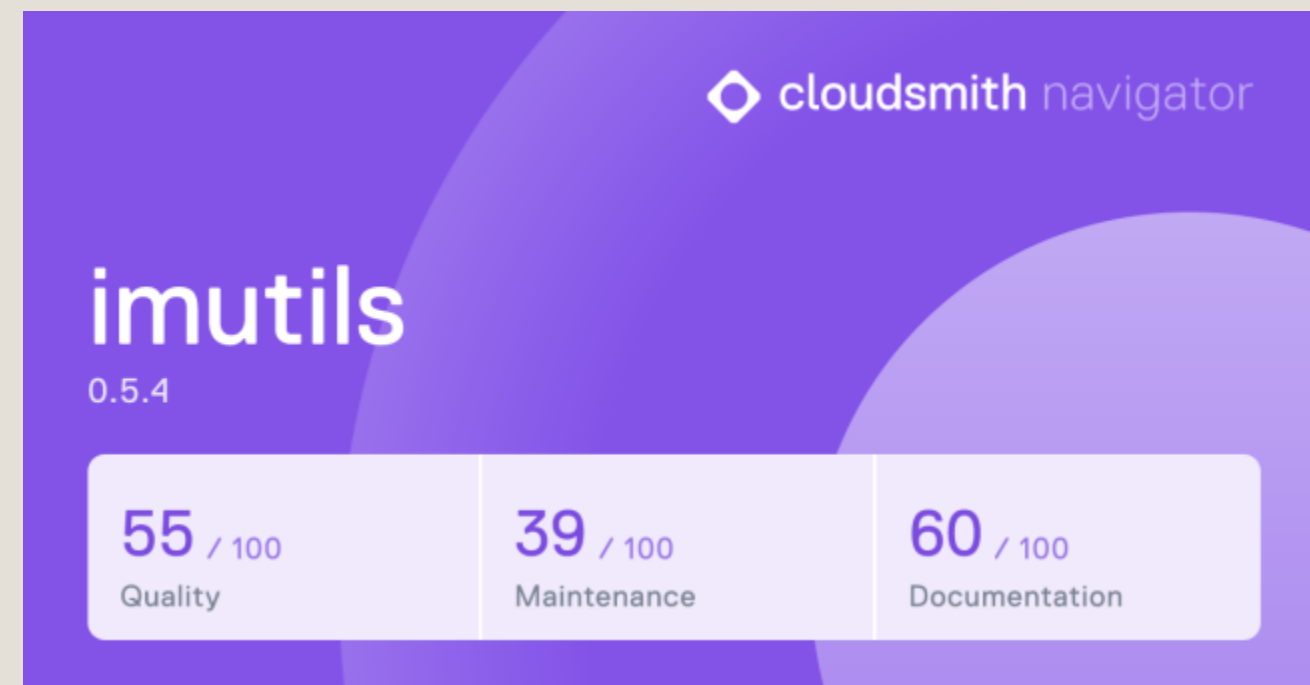
Visualizer



트레이너



실행기



발표

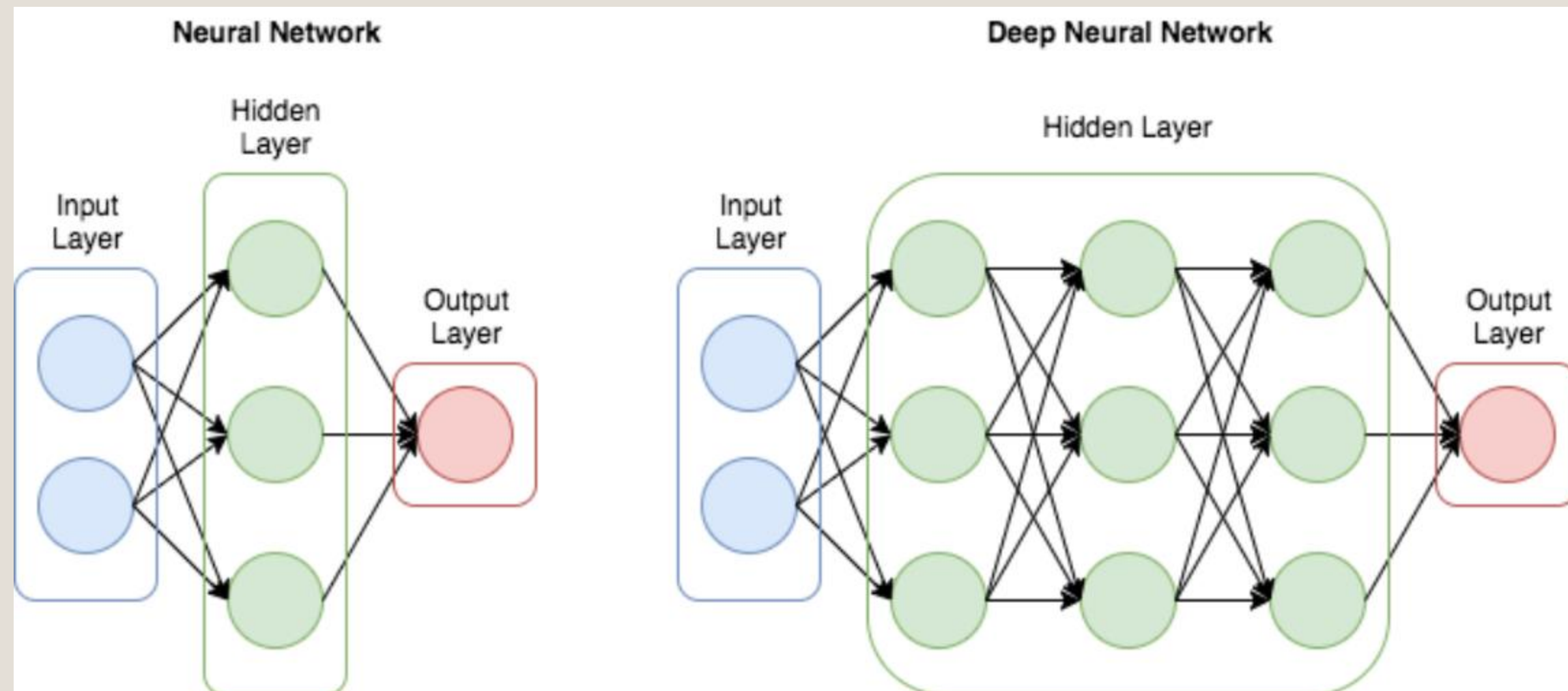
실행기 설명

그리고 DNN

트레이너



실행기



발표

실행기 설명

Visualizer

트레이너



실행기



```
from tensorflow.keras.applications.mobilenet_v2 import preprocess_input
from tensorflow.keras.preprocessing.image import img_to_array
from tensorflow.keras.models import load_model
from imutils.video import VideoStream
import numpy as np
import imutils
import time
import cv2
import matplotlib.pyplot as plt

# 외부 카메라와 연결시 인풋력으로 인한 깨짐현상 있어서 강제로 값을 넣어줌
prev_time = 0
FPS = 30

# Case Similarity를 저장할 리스트
safetyhelmet_similarities = []

def detect_and_predict_hat(frame, faceNet, hatNet):
    # 프레임의 치수 잡기
    (h, w) = frame.shape[:2]
    blob = cv2.dnn.blobFromImage(frame, 1.0, (224, 224), (104.0, 177.0, 123.0))

    # 얼굴 감지
    faceNet.setInput(blob)
    detections = faceNet.forward()

    # 얼굴 목록, 해당 위치 및 얼굴 마스크 네트워크의 예측 목록을 초기화
    faces = []
    locs = []
    preds = []

    # 탐지 ing
    for i in range(0, detections.shape[2]):
        # 확률 추출
        confidence = detections[0, 0, i, 2]
        # 작은 확률 버리기
        if confidence > 0.5:
            # 객체에 대한 경계 상자의 (x, y) 좌표를 계산
            box = detections[0, 0, i, 3:7] * np.array([w, h, w, h])
            (startX, startY, endX, endY) = box.astype("int")

            # 박스 크기
            (startX, startY) = (max(0, startX), max(0, startY))
            (endX, endY) = (min(w - 1, endX), min(h - 1, endY))

            # 카메라
            face = frame[startY:endY, startX:endX]
            face = cv2.cvtColor(face, cv2.COLOR_BGR2RGB)
            face = cv2.resize(face, (224, 224))
            face = img_to_array(face)
            face = preprocess_input(face)

            faces.append(face)
            locs.append((startX, startY, endX, endY))
```

```
# 얼굴만 나왔을때 예측
if len(faces) > 0:
    faces = np.array(faces, dtype="float32")
    preds = hatNet.predict(faces, batch_size=32)

    return (locs, preds)

# cv2 (DNN)뉴런 모듈 사용
prototxtPath = r"face_detector\deploy.prototxt"
weightsPath = r"face_detector\res10_300x300_ssd_iter_140000.caffemodel"
faceNet = cv2.dnn.readNet(prototxtPath, weightsPath)

hatNet = load_model("mask_detector.model")

print("[INFO] starting video stream...")
vs = VideoStream(src=0).start()

while True:
    # 프레임 조절할 것
    frame = vs.read()
    ret = vs.read()
    current_time = time.time() - prev_time
    if (ret is True) and (current_time > 1. / FPS):
        prev_time = time.time()
        cv2.imshow('VideoCapture', frame)

        if cv2.waitKey(1) > 0:
            break

    frame = imutils.resize(frame, width=1024)
    (locs, preds) = detect_and_predict_hat(frame, faceNet, hatNet)

    for (box, pred) in zip(locs, preds):
        (startX, startY, endX, endY) = box
        (hat, withoutthat) = pred

        # Safetyhelmet 케이스에 대한 유사도 계산
        safetyhelmet_similarity = 1.0 - withoutthat

        # 얼굴 주위에 수치 표시
        text = f"Similarity: {safetyhelmet_similarity:.2f}"
        cv2.putText(frame, text, (startX, startY - 60), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255, 255, 255), 2)

        # Safetyhelmet 케이스 유사도 임계값 설정
        baseline = 0.7

        label = "Please put on your Hardhat" if safetyhelmet_similarity < baseline else "PASS"
        color = (0, 255, 0) if label == "PASS" else (0, 0, 255)

        cv2.putText(frame, label, (startX, startY - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.45, color, 2)
        cv2.rectangle(frame, (startX, startY), (endX, endY), color, 2)

    # Safetyhelmet 유사도를 리스트에 추가
    safetyhelmet_similarities.append(safetyhelmet_similarity)
```

발표

실행기 설명

Visualizer

```
        if key == ord("q"):
            break

# Case Similarity 평균값을 파일에 저장
with open("similarity.txt", "w") as file:
    for cs in safetyhelmet_similarities:
        file.write(f"{cs}\n")

# 파일에서 similarity 데이터 읽어오기
with open("similarity.txt", "r") as file:
    similarity_data = [float(line.strip()) for line in file]

# 그래프 생성 및 이미지로 저장
plt.plot(similarity_data)
plt.xlabel('Frame')
plt.ylabel('Safetyhelmet Similarity')
plt.title('Safetyhelmet Over Frames')
plt.savefig('sim.png')
plt.show()

cv2.destroyAllWindows()
vs.stop()
```

트레이너



실행기



이름	수정된 날짜	유형	크기
dataset	2023-03-21 오후 4:33	파일 폴더	
face_detector	2023-03-22 오후 1:58	파일 폴더	
model	2023-06-13 오후 10:08	파일 폴더	
module	2023-06-13 오후 10:15	파일 폴더	
ddd	2023-03-21 오후 5:09	JetBrains PyChar...	1KB
mask_detector.model	2023-11-13 오전 12:01	MODEL 파일	11,230KB
plot	2023-11-13 오전 12:01	PNG 파일	27KB
sim	2023-11-13 오전 1:05	PNG 파일	53KB
similarity	2023-11-13 오전 1:05	텍스트 문서	3KB
test	2023-11-13 오전 1:04	JetBrains PyChar...	5KB
train_mask_detector	2023-11-14 오전 2:33	JetBrains PyChar...	4KB

발표

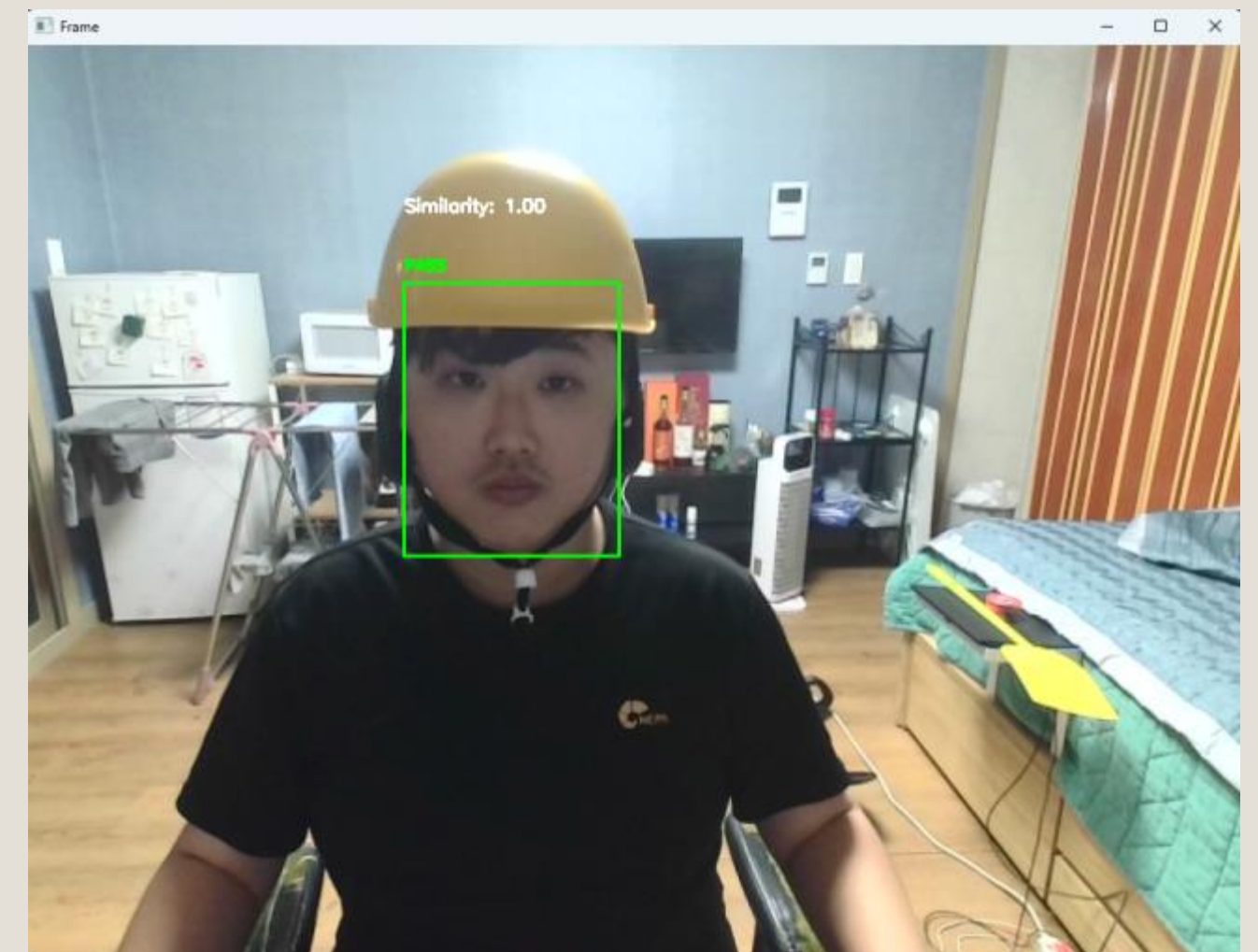
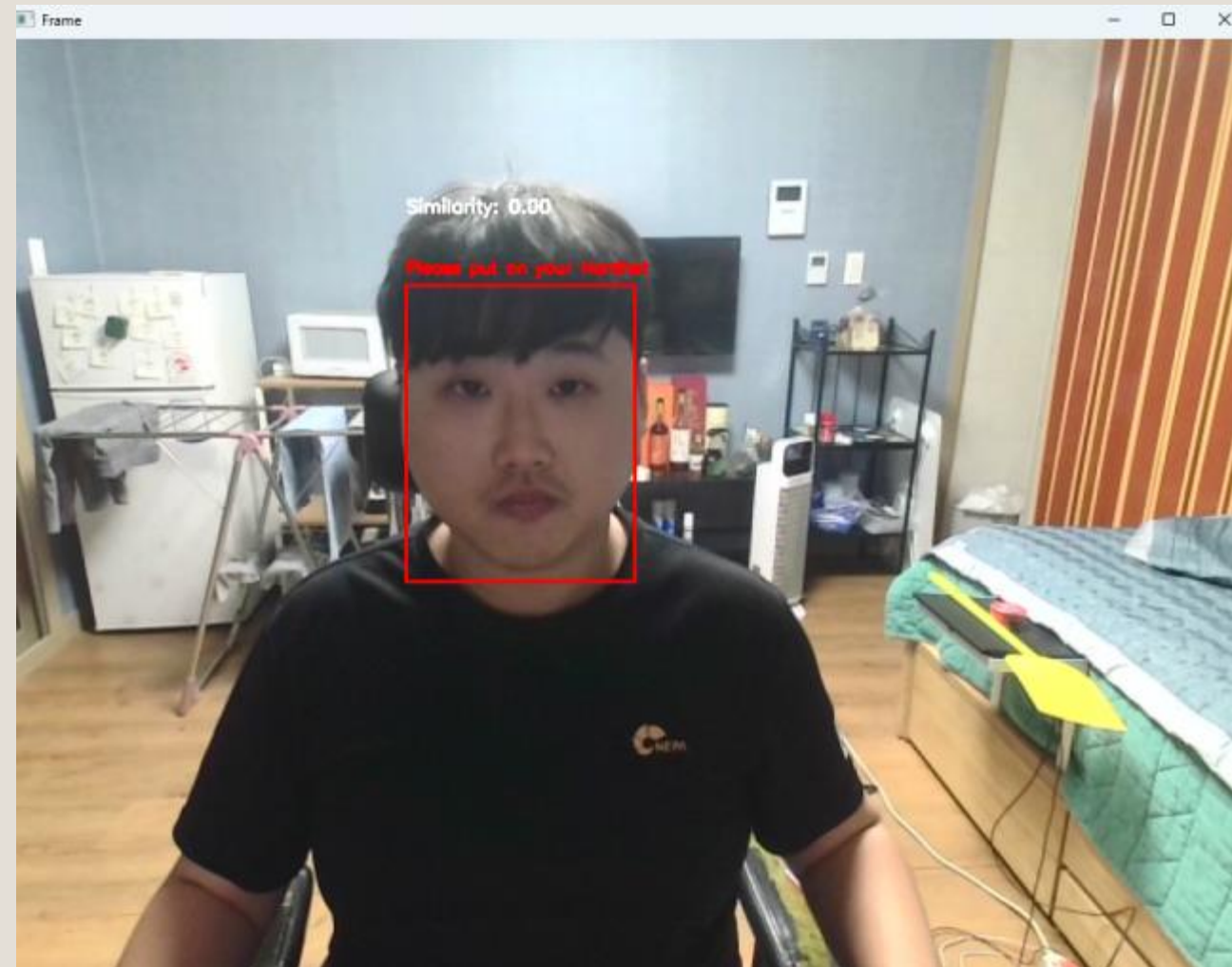
실행기 설명

Visualizer

트레이너



실행기



발표

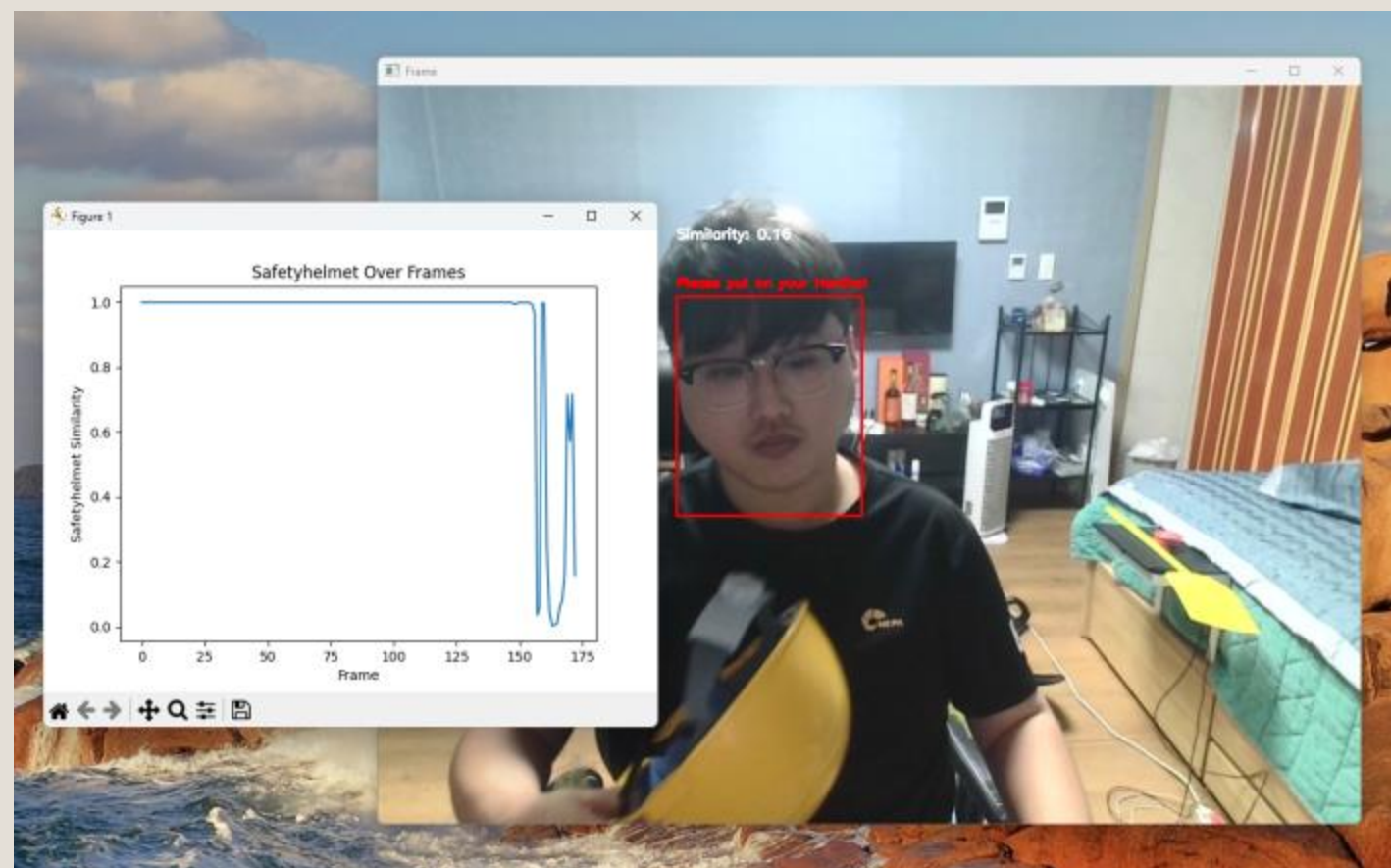
실행기 설명

Visualizer

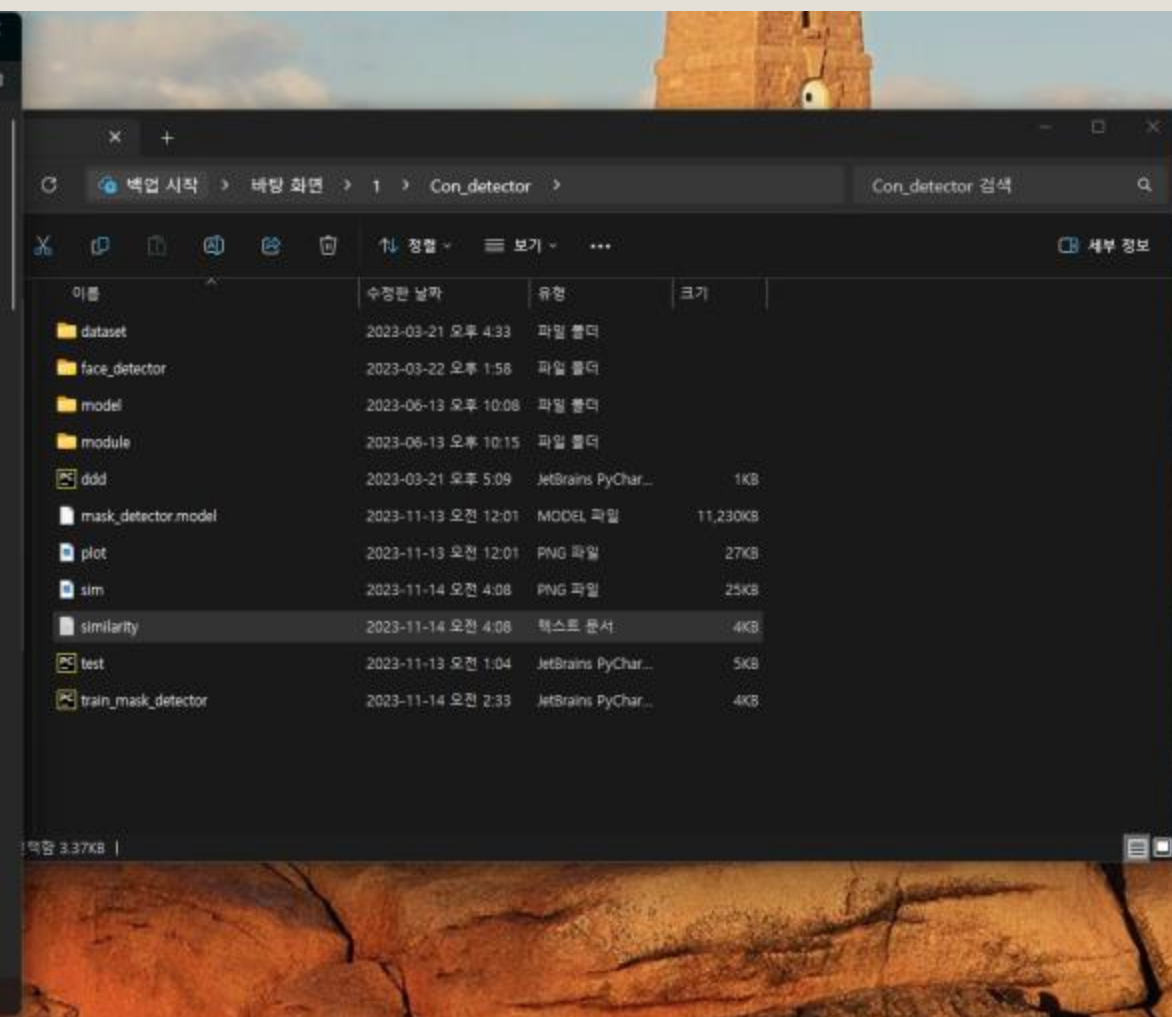
트레이너



실행기



```
similarity
파일 편집 보기
0.9999999456415054
0.9999999889753646
0.9999999372384636
0.99999995218582285
0.999999952981069
0.9999999188893653
0.999999946529892
0.9999987601644307
0.9999987601644307
0.9999999851172587
0.9999999851172587
0.9999999741978218
0.9999997645148397
0.9999999848877916
0.9999998928745271
0.999999889832097
0.9999999797035155
0.9999999817101806
0.9999999817101806
0.9999999685913998
0.999999932684576
0.99999998350971
0.9999999843364211
0.999999927225365
0.9999999656378371
0.9999987913192
0.9999999847717618
0.9999999629837895
0.9999999520375802
0.9999999646913444
0.999998457284249
0.999998457284249
0.9999999560123847
0.9999999560123847
0.999999989008538
0.9999999687796312
0.9999999877830543
0.9999999877830543
0.9999999166391547
0.999996652265736
줄 1, 열 1
```



결론

Conclusion

결론, 맺음말, 소감

와! 끝났다!

01

위기를
기회로!

Kaggle이 동작하지 않았기 때문에
재밌는 경험 추가

02

재밌었지만
아쉬웠다

결과가 성공적으로 출력되어 만족하나
분류가 정확하지 않을 때가 간혹 있었다
코드 추가로 결과값은 잡았으나 아쉬움

03

조금 더
해볼까?

(여유 있을 때) 해당 코드에서 아쉬운 점을
보완해 제대로 된 결과를 내겠다

Make Experience Great Again

질문 있을까요?

Q&A

질문
있을까요?

감사합니다!

THE
END