**프로그램 기본 코드구현**

**김태호**

**기능 1. 설치된 카메라에서 얼굴 사진을 서버로 보내기.**

**고려사항.**

1. 카메라가 아무 사진이나 계속해서 서버로 보내면 안됨.

* 해결방안: 얼굴을 인식하여 일정 시간 이상 지속되면 서버로 보냄.

1. 여러 각도의 사진이 여러 장 있어야 인식 정확도가 올라감.

* 해결방안: 특정 프레임마다 계속하여 캡쳐 하여 서버로 보냄.

1. 사진을 찍는 과정은 서버에서 하는 것이 아니기 때문에 얼굴인식 과정에서 무거운 모델이나 알고리즘을 돌릴 수 없음.

* 해결방안: 이 과정은 딥러닝이 아닌 OpenCV의 간단한 CV기술로 구현함.

**<Main Code>**

**import** **cv2**

**import** **datetime**

cap = cv2.VideoCapture(0);

*# 이미지를 캡쳐한 시간을 이름으로 저장함.*

now = datetime.datetime.now().strftime("%y-%m-**%d**\_%H-%M-%S")

width = cap.get(cv2.CAP\_PROP\_FRAME\_WIDTH)

height = cap.get(cv2.CAP\_PROP\_FRAME\_HEIGHT)

green = (0, 255, 0)

magenta = (255, 0, 255)

location = (0, 100)

font = cv2.FONT\_ITALIC

temp = 1

print("size: **{0}** x **{1}**".format(width, height))

*# 영상 저장을 위한 VideoWriter 인스턴스 생성*

fourcc = cv2.VideoWriter\_fourcc(\*'XVID')

*#얼굴 인식용*

face\_cascade = cv2.CascadeClassifier()

face\_cascade.load('haarcascade\_frontalface\_default.xml')

**while**(**True**):

*#캠으로 부터 이미지 얻기*

ret, frame = cap.read()

*#더 이상 이미지가 없으면 종료*

**if** frame **is** **None**:

**break**;

*# 얼굴인식 영상 처리*

grayframe = cv2.cvtColor(frame, cv2.COLOR\_BGR2GRAY)

blur = cv2.GaussianBlur(grayframe,(5,5), 0)

faces = face\_cascade.detectMultiScale(blur, 1.8, 2, 0, (50, 50))

*# 얼굴인식 여부 확인*

**if** len(faces) == 0:

temp = 1

**else**:

temp += 1

*# 얼굴이 4프레임 이상 인식되면 4프레임마다 캡쳐*

**if** temp%4 == 0:

print("캡쳐")

cv2.imwrite("image/capture/" + str(now) + ".png", frame)

*# 얼굴인식여부를 화면상에 표시*

**if** temp < 4:

cv2.putText(frame,"I can't recognize your face.", location, font, 1, green, 2)

**else**:

cv2.putText(frame,"I'm checking it now.", location, font, 1, magenta, 2)

*# 얼굴 인식된 이미지 화면 표시*

cv2.imshow('face\_detect',frame)

*# q 를 누르면 종료*

**if** cv2.waitKey(1) & 0xFF == ord('q'):

**break**

cap.release()

cv2.destroyAllWindows()

**결과**

* **얼굴을 인식하지 못했을 경우**



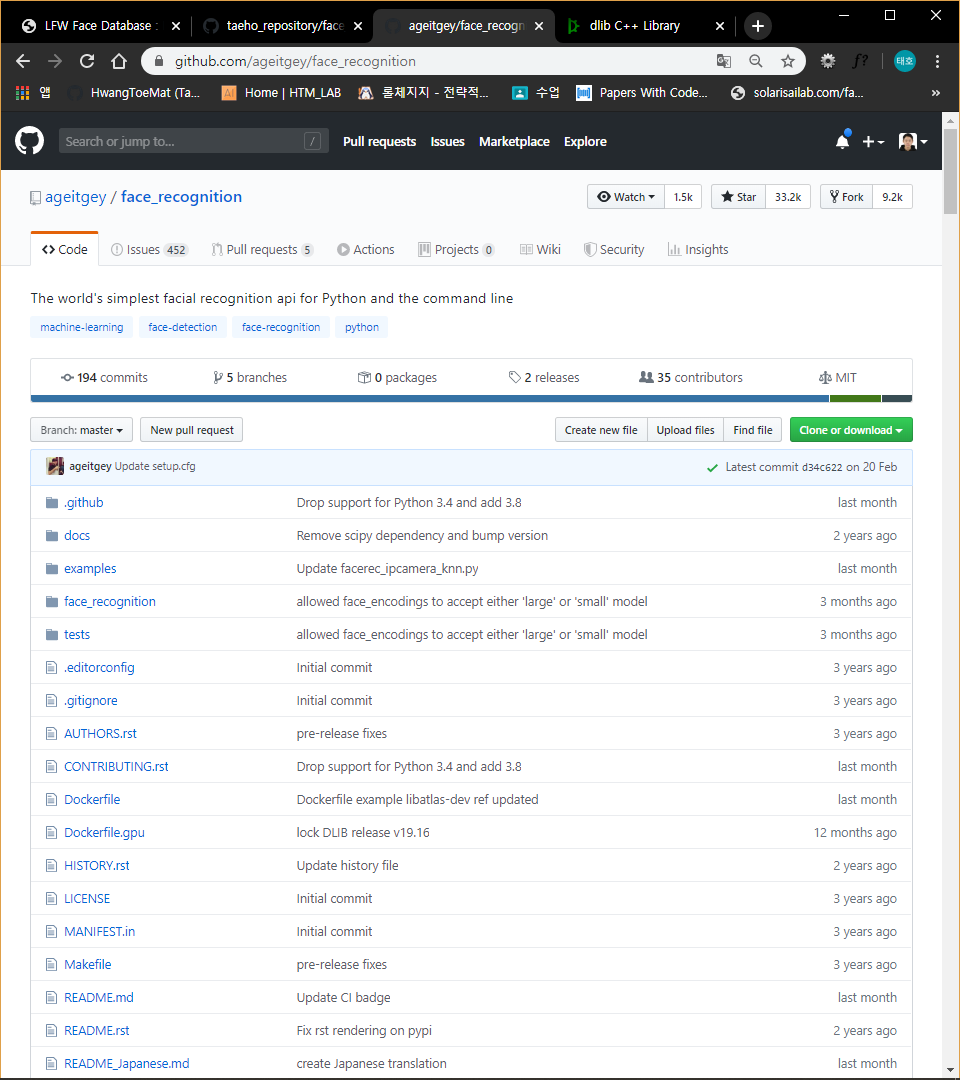
* **얼굴을 인식했을 경우(캡쳐하여 저장함.)**



**문제점 인식**

1. **얼굴인식과정에서 정확도가 떨어지는 모습을 보임(얼굴을 인식하는데 시간이 걸리거나 다른 사물을 보고 가끔 얼굴로 인식함.)**

**기능 2. 얼굴인식이 구현된 라이브러리 이용해 보기. (Face Recognition)**



* (Github link : <https://github.com/ageitgey/face_recognition>)

**소개**

* 33만개 이상의 star를 받은 라이브러리임.
* 딥러닝 기반으로 제작된 dlib의 얼굴 인식 기능을 사용하여 구축됨.
* 이 모델은 Labeled Faces in the Wild(LFW)기준으로 99.38%의 정확도를 가짐.
* 이미지에서 얼굴을 찾고(detection), 얼굴의 feature를 찾는 과정(extraction)은 이미 학습되어 있기 때문에 추가로 학습시킬 필요가 없음.
* 따라서 이미지에 레이블을 지정하여 입력하면 그 이미지에서 나은 feature들을 feature space로 임베딩하여 거리에 따라 데이터를 구분함.
* 모델 전체에 대한 재학습이 필요 없음.
* 모델구현에 관한 여러 예시가 공개되어 있고 세세한 파라미터들을 직접 설정 가능하게함.

**<Main Code>**

**=> 이미지 한장으로 학습시키기**

**import** **face\_recognition**

**import** **cv2**

**import** **numpy** **as** **np**

*# Get a reference to webcam #0 (the default one)*

video\_capture = cv2.VideoCapture(0)

*# Load a sample picture and learn how to recognize it.*

jeongsue\_image = face\_recognition.load\_image\_file("image/face\_recognition/정수.jpg")

jeongsue\_face\_encoding = face\_recognition.face\_encodings(jeongsue\_image)[0]

*# Load a second sample picture and learn how to recognize it.*

taeho\_image = face\_recognition.load\_image\_file("image/face\_recognition/태호.jpg")

taeho\_face\_encoding = face\_recognition.face\_encodings(taeho\_image)[0]

*# Load a second sample picture and learn how to recognize it.*

donghyeok\_image = face\_recognition.load\_image\_file("image/face\_recognition/동혁.jpg")

donghyeok\_face\_encoding = face\_recognition.face\_encodings(donghyeok\_image)[0]

*# Create arrays of known face encodings and their names*

known\_face\_encodings = [

jeongsue\_face\_encoding,

taeho\_face\_encoding,

donghyeok\_face\_encoding

]

known\_face\_names = [

"JeongSue",

"TaeHo",

"DongHyeok"

]

*# Initialize some variables*

face\_locations = []

face\_encodings = []

face\_names = []

process\_this\_frame = **True**

**while** **True**:

*# Grab a single frame of video*

ret, frame = video\_capture.read()

*# Resize frame of video to 1/4 size for faster face recognition processing*

small\_frame = cv2.resize(frame, (0, 0), fx=0.25, fy=0.25)

*# Convert the image from BGR color (which OpenCV uses) to RGB color (which face\_recognition uses)*

rgb\_small\_frame = small\_frame[:, :, ::-1]

*# Only process every other frame of video to save time*

**if** process\_this\_frame:

*# Find all the faces and face encodings in the current frame of video*

face\_locations = face\_recognition.face\_locations(rgb\_small\_frame)

face\_encodings = face\_recognition.face\_encodings(rgb\_small\_frame, face\_locations)

face\_names = []

**for** face\_encoding **in** face\_encodings:

*# See if the face is a match for the known face(s)*

matches = face\_recognition.compare\_faces(known\_face\_encodings, face\_encoding)

name = "Unknown"

*# Or instead, use the known face with the smallest distance to the new face*

face\_distances = face\_recognition.face\_distance(known\_face\_encodings, face\_encoding)

best\_match\_index = np.argmin(face\_distances)

min\_val = min(face\_distances)

**if** matches[best\_match\_index]:

**if** min\_val < 0.6:

name = known\_face\_names[best\_match\_index]

face\_names.append(name)

process\_this\_frame = **not** process\_this\_frame

*# Display the results*

**for** (top, right, bottom, left), name **in** zip(face\_locations, face\_names):

*# Scale back up face locations since the frame we detected in was scaled to 1/4 size*

top \*= 4

right \*= 4

bottom \*= 4

left \*= 4

*# Draw a box around the face*

cv2.rectangle(frame, (left, top), (right, bottom), (0, 0, 255), 2)

*# Draw a label with a name below the face*

cv2.rectangle(frame, (left, bottom - 35), (right, bottom), (0, 0, 255), cv2.FILLED)

font = cv2.FONT\_HERSHEY\_DUPLEX

cv2.putText(frame, name, (left + 6, bottom - 6), font, 1.0, (255, 255, 255), 1)

*# Display the resulting image*

cv2.imshow('Video', frame)

*# Hit 'q' on the keyboard to quit!*

**if** cv2.waitKey(1) & 0xFF == ord('q'):

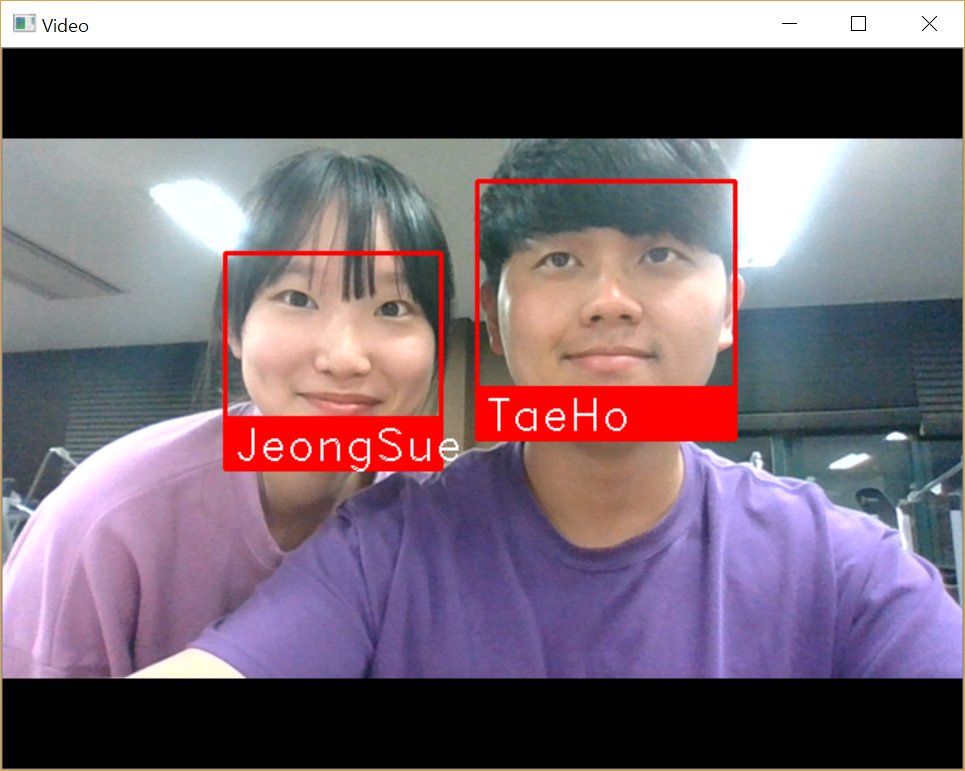
**break**

*# Release handle to the webcam*

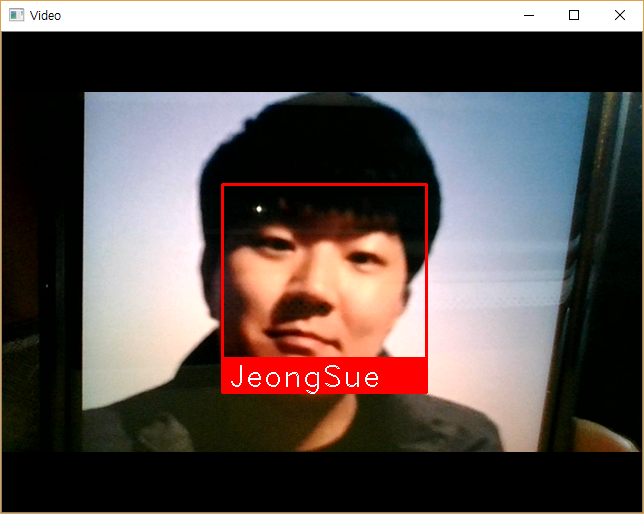
video\_capture.release()

cv2.destroyAllWindows()

**결과**

****

**문제점 인식**

****

등록되지 않았더라도 너무 닮은 사람은 Unknown으로 인식하지 못함. (이 문제는 입력 데이터가 너무 적고 레이블의 수도 적어서 그렇다고 생각함. 더 많은 사람에 대해 학습시키면 이러한 문제가 줄어들 것으로 보임.)