import numpy as np

import cv2

import dlib

faceCascade = cv2.CascadeClassifier('haarcascade\_frontalcatface.xml')

predictor = dlib.shape\_predictor("shape\_predictor\_68\_face\_landmarks.dat")

JAWLINE = list(range(0,17))

RIGHT\_EYEBROW = list(range(17,22))

LEFT\_EYEBROW = list(range(22,27))

NOSE = list(range(27,36))

RIGHT\_EYE = list(range(36,42))

LEFT\_EYE = list(range(42,48))

MOUTH\_OUTLINE = list(range(48,61))

MOUTH\_INNER = list(range(61,68))

def detect(gray, frame):

faces = faceCascade.detectMultiScale(gray, scaleFactor=1.05, minNeighbors=5, minSize=(100,100), flags=cv2.CASCADE\_SCALE\_IMAGE)

for (x,y,w,h) in faces:

dlib\_rect = dlib.rectangle(int(x), int(y), int(x+w), int(y+h))

landmarks = np.matrix([[p.x, p.y] for p in predictor(frame, dlib\_rect).parts()])

landmarks\_display = landmarks[0:68]

for idx, point in enumerate(landmarks\_display):

pos = (point[0, 0], point[0, 1])

cv2.circle(frame, pos, 2, color=(0,255,255), thickness=-1)

return frame

video\_capture = cv2.VideoCapture(0)

while True:

\_, frame = video\_capture.read()

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

canvas = detect(gray, frame)

cv2.imshow("window", canvas)

cv2.waitKey(0)

video\_capture.release()

cv2.destroyAllWindows()

* 위에거가찐코드야~ㅎㅎㅎㅎ

import numpy as np

import cv2

import dlib

faceCascade = cv2.CascadeClassifier('haarcascade\_frontalcatface.xml')

predictor = dlib.shape\_predictor("shape\_predictor\_68\_face\_landmarks.dat")

ser = 0

JAWLINE = list(range(0,17))

RIGHT\_EYEBROW = list(range(17,22))

LEFT\_EYEBROW = list(range(22,27))

NOSE = list(range(27,36))

RIGHT\_EYE = list(range(36,42))

LEFT\_EYE = list(range(42,48))

MOUTH\_OUTLINE = list(range(48,61))

MOUTH\_INNER = list(range(61,68))

def detect(gray, frame):

faces = faceCascade.detectMultiScale(gray, scaleFactor=1.05, minNeighbors=5, minSize=(100,100), flags=cv2.CASCADE\_SCALE\_IMAGE)

for (x,y,w,h) in faces:

dlib\_rect = dlib.rectangle(int(x), int(y), int(x+w), int(y+h))

landmarks = np.matrix([[p.x, p.y] for p in predictor(frame, dlib\_rect).parts()])

landmarks\_display = landmarks[0:68]

if landmarks[36:48] == 0:

ser = serial.Serial('COM6', 9600)

print(ser)

ser.write('SLEEP')

else:

ser = serial.Serial('COM6', 9600)

print(ser)

ser.write('AWAKE')

for idx, point in enumerate(landmarks\_display):

pos = (point[0, 0], point[0, 1])

cv2.circle(frame, pos, 2, color=(0,255,255), thickness=-1)

return frame

video\_capture = cv2.VideoCapture(0)

while True:

\_, frame = video\_capture.read()

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

canvas = detect(gray, frame)

cv2.imshow("window", canvas)

cv2.waitKey(0)

video\_capture.release()

cv2.destroyAllWindows()