import cv2

import dlib

import numpy as np

# Load pre-trained models for age and gender prediction

age\_net = cv2.dnn.readNetFromCaffe('models/age\_deploy.prototxt', 'models/age\_net.caffemodel')

gender\_net = cv2.dnn.readNetFromCaffe('models/gender\_deploy.prototxt', 'models/gender\_net.caffemodel')

# Initialize dlib's face detector

detector = dlib.get\_frontal\_face\_detector()

# Pre-defined labels

age\_labels = ['(0-5)', '(6-10)', '(11-15)', '(16-20)', '(21-27)', '(28-35)', '(36-45)', '(46-60)''(60-65)', '(66-90)']

gender\_labels = ['Male', 'Female']

def predict\_age\_gender(face\_image):

blob = cv2.dnn.blobFromImage(face\_image, 1.0, (227, 227), (78.4263377603, 87.7689143744, 114.895847746),

swapRB=False)

# Predict gender

gender\_net.setInput(blob)

gender\_preds = gender\_net.forward()

gender = gender\_labels[gender\_preds[0].argmax()]

# Predict age

age\_net.setInput(blob)

age\_preds = age\_net.forward()

age = age\_labels[age\_preds[0].argmax()]

return age, gender

def detect\_and\_predict(image\_path):

image = cv2.imread(image\_path)

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

faces = detector(gray)

for face in faces:

x, y, w, h = face.left(), face.top(), face.width(), face.height()

face\_image = image[y:y + h, x:x + w]

face\_image = cv2.resize(face\_image, (227, 227)) # Resize to match model input size

age, gender = predict\_age\_gender(face\_image)

label = f"{gender}, {age}"

cv2.rectangle(image, (x, y), (x + w, y + h), (0, 255, 0), 2)

cv2.putText(image, label, (x, y - 10), cv2.FONT\_HERSHEY\_SIMPLEX, 0.5, (0, 255, 0), 2)

return image

if \_\_name\_\_ == "\_\_main\_\_":

image\_path = r"C:\Users\DISPATCHER\Desktop\BCE 211\images\1.jpg"

output\_image = detect\_and\_predict(image\_path)

cv2.imshow("Output", output\_image)

cv2.waitKey(0)

cv2.destroyAllWindows()



