import cv2

import numpy as np

from tensorflow.keras.preprocessing import image

# Load pre-trained face detection and emotion models

face\_cascade = cv2.CascadeClassifier('haarcascade\_frontalface\_default.xml')

emotion\_model = tf.keras.models.load\_model('facial\_emotion\_recognition.h5')

# Defining emotion categories

emotions = ["angry", "disgust", "fear", "happy", "neutral", "sad", "surprise"]

def detect\_and\_classify\_emotions (https://github.com/edward302/Assignment-2/blob/main/20240529\_153201.jpg):

"""Detects faces in an image and classifies their emotions.

Args:

image\_path: https://github.com/edward302/Assignment-2/blob/main/20240529\_153201.jpg

Returns:

A list of dictionaries, where each dictionary contains:

- x1: Top-left x-coordinate of the bounding box.

- y1: Top-left y-coordinate of the bounding box.

- x2: Bottom-right x-coordinate of the bounding box.

- y2: Bottom-right y-coordinate of the bounding box.

- emotion: Predicted emotion label.

"""

img = cv2.imread (<https://github.com/edward302/Assignment-2/blob/main/20240529_153201.jpg>)

# Read the image

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

# Convert to grayscale

# Detect faces

faces = face\_cascade.detectMultiScale(gray, 1.1, 4)

# Process each detected face

face\_data = []

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

cv2.rectangle(img, (x, y), (x+w, y+h), (255, 0, 0), 2) # Draw bounding box

# Extract face region of interest (ROI)

roi = gray[y:y+h, x:x+w]

roi = cv2.resize(roi, (48, 48)) # Resize to model's input size

roi = roi.astype("float") / 255.0 # Normalize pixel values

roi = np.expand\_dims(roi, axis=0) # Add a dimension for batch processing

# Predict emotion

prediction = emotion\_model.predict(roi)[0] # Get the probability vector

emotion\_label = emotions[np.argmax(prediction)] # Get predicted emotion

# Store face data

face\_data.append({

"x1": x,

"y1": y,

"x2": x + w,

"y2": y + h,

"emotion": emotion\_label

})

return face\_data

# usage

image\_path = <https://github.com/edward302/Assignment-2/blob/main/20240529_153201.jpg>

face\_data = detect\_and\_classify\_emotions(https://github.com/edward302/Assignment-2/blob/main/20240529\_153201.jpg)

if face\_data:

for face in face\_data:

print(f"Face coordinates: ({face['x1']}, {face['y1']}), ({face['x2']}, {face['y2']})")

print(f"Predicted emotion: {face['emotion']}")

else:

print("No faces detected in the image.")