README

opencv检测人脸(效果不佳)

原理：用opencv自带库去检测人脸，然后根据人脸矩形宽高比来判断是否是正脸

预测脚本

```
import cv2
import numpy as np
import os

def detect_frontal_face(img_path,save_path):
    img = cv2.imread(img_path)
    """检测人脸并判断是否为正脸"""
    # 灰度化图片
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

    # 加载haarface级联分类器
    face_cascade =
cv2.CascadeClassifier(r'D:\anaconda3\envs\py38\Library\etc\haarcascades\haarcascade_f
rontalface_default.xml')

    # 检测人脸
    faces = face_cascade.detectMultiScale(gray, 1.3, 5)
    if len(faces)>=1:
        # 在原图img上画矩形
        for (x, y, w, h) in faces:
            # 获取人脸矩形区域
            face_img = img[y:y+h, x:x+w]
            cv2.imshow('Face Detection', face_img)
            cv2.waitKey(0)
            # # 判断人脸矩形宽高比是否为正脸
            if w / h < 1.2 and w / h > 0.8:
                cv2.rectangle(img, (x, y), (x+w, y+h), (0, 255, 0), 2)
                label = 'Frontal Face'
            else:
                cv2.rectangle(img, (x, y), (x+w, y+h), (0, 0, 255), 2)
                label = 'Not Frontal'

            # 在图片上打标签
            cv2.putText(img, label, (x, y - 5),cv2.FONT_HERSHEY_SIMPLEX, 0.6, (0,
```

```
255, 255), 2)

        # 展示图片
        # cv2.imshow('Face Detection', img)
        # cv2.waitKey(0)
        cv2.imwrite(save_path,img)
    else:
        # cv2.imwrite(save_path, img)
        print("No facial information was detected")

dir_path = r'E:\pythonProject\xznsh\img'
save_path = r'E:\pythonProject\xznsh\img_result'
imgs = os.listdir(dir_path)
# print(imgs)
for img in imgs:
    img_path = os.path.join(dir_path, img)
    # 判断是否为图片文件
    if img_path.endswith('.jpg') or img_path.endswith('.png'):
        detect_frontal_face(img_path,os.path.join(save_path,"result_"+img))
```

误检示例

唯一检测正确示例


2023年05月09日 星期二
Frontal Face
营造优美舒适环境
倡导文明健康生活
泉河支行自助外厅环境1