# 颜值测试仪（摄像头版）

案例描述：获取摄像头图片，调用百度AI进行识别。

本范例的具体介绍请参考百度AI的文档。

https://ai.baidu.com/docs#/Face-Python-SDK/81dd3e06

# 准备工作

1.导入库

In [ ]:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
import cv2
import time
import base64
from aip import AipFace
from IPython import display
from matplotlib import pyplot as plt
import matplotlib
%matplotlib inline
```

2.定义变量

In [ ]:

```
face_num = 0
frame = None
now_time = 0
```

3.设置认证信息

注：这里用的是测试账号，有访问次数的限制，请使用自己的账号信息。

In [ ]:

```
""" 你的 APPID AK SK """
APP_ID = '15469649'
API_KEY = '3vZgLINSnGGEafPflkTLzkGh'
SECRET_KEY = '8cUXtkMed2z86kqfyrV606ylnCmfcc48'
client = AipFace(APP_ID, API_KEY, SECRET_KEY)
imageType = "BASE64"
options = {}
options["face_field"] = "age,beauty,expression,gender,glasses"
options["max_face_num"] = 2
options["face_type"] = "LIVE"
options["liveness_control"] = "LOW"
```

4.基本函数：读取图片

In [ ]:

```
def cvimg_to_b64(img):
    try:
        image = cv2.imencode('.jpg', img)[1]
        base64_data = str(base64.b64encode(image))[2:-1]
        return base64_data
    except Exception as e:
        return "error"
```

5.基本函数：框出人脸

In [ ]:

```
#注意：haarcascade_frontalface_default.xml要放在同一个文件夹下。
def faceDetect(img, face_cascade=cv2.CascadeClassifier('haarcascade_frontalface_defa
    size = img.shape[:2]
    divisor = 8
    h, w = size
    minSize = (w // divisor, h // divisor)
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = face_cascade.detectMultiScale(gray, 1.2, 1, cv2.CASCADE_SCALE_IMAGE, mi
    for (x, y, w, h) in faces:
        cv2.rectangle(img, (x, y), (x + w, y + h), (255, 0, 0), 2)
    return img, len(faces)
```

6.基本函数：将信息写到图片

In [ ]:

```
#将text写到图片上
def put_Text(cvimg, text, location, size=2):
    cvimg = cv2.putText(cvimg, text, location, cv2.FONT_HERSHEY_SIMPLEX, size, (51,
    return cvimg
```

# 开始工作

描述：摄像头将拍摄照片，并上传到百度AI平台进行识别，然后将识别结果输出来。

In [ ]:

```
cap = cv2.VideoCapture(0)
ret, frame = cap.read()
if ret:
    frame1, face_num = faceDetect(frame)
    frame1 = cv2.flip(frame1, 1, dst=None)
    frame1 = cv2.resize(frame1, (1280, 800), interpolation=cv2.INTER_LINEAR)
    img64 = cvimg_to_b64(frame1)
    #获取图片信息
    res = client.detect(img64, imageType, options)
    #如果找到人脸信息就读出
    if (res['error_msg']=="SUCCESS"):
        gender = res['result']['face_list'][0]['gender']['type']
        age = res['result']['face_list'][0]['age']
        beauty = res['result']['face_list'][0]['beauty']
        frame1 = put_Text(frame1, str(age), (300, 50))
        frame1 = put_Text(frame1, str(gender), (300, 120))
        frame1 = put_Text(frame1, str(beauty), (300, 190))
        frame1 = put_Text(frame1, "Age:", (50, 50))
        frame1 = put_Text(frame1, "Gender:", (50, 120))
        frame1 = put_Text(frame1, "Beauty:", (50, 190))
    else:
        frame1 = put_Text(frame1, "no face", (50, 50))
    #display.clear_output(wait=True)
    img=frame1[:,:,::-1]
    plt.imshow(img)
    plt.axis('off') #不显示坐标
    plt.show()
else:
    print("没有接摄像头或者摄像头被占用！")
cap.release()
cv2.destroyAllWindows()
```

## ## 综合拓展

功能描述：当摄像头前面有障碍（有人），LED亮起，自动拍摄照进行识别。识别结束后，LED熄灭，显示识别结果，图片自动保存，。

装置搭建：红外测障传感器接在D3脚；舵机接到D7；LED接到D13。

其他说明：请设计一张颜值指示表，并测试舵机的指向情况。

In [ ]:

```
#注意事项：测试下面的代码，每一次运行都要先通过“服务-重启 & 清空输出”来初始化。

from xugu import Pin,Servo
p1 = Pin(3, Pin.IN)
led = Pin(13, Pin.OUT)
servo = Servo(7)
while True:
    v1=p1.read_digital()
    if v1==1:
        led.write_digital(1)
        print("开始测试，请稍候")
        cap = cv2.VideoCapture(0)
        ret, frame = cap.read()
        if ret:
            frame1, face_num = faceDetect(frame)
            frame1 = cv2.flip(frame1, 1, dst=None)
            frame1 = cv2.resize(frame1, (1280, 800), interpolation=cv2.INTER_LINEAR)
            img64 = cvimg_to_b64(frame1)
            #获取图片信息
            res = client.detect(img64, imageType, options)
            #如果找到人脸信息就读出
            if (res['error_msg']=="SUCCESS"):
                gender = res['result']['face_list'][0]['gender']['type']
                age = res['result']['face_list'][0]['age']
                beauty = res['result']['face_list'][0]['beauty']
                frame1 = put_Text(frame1, str(age), (300, 50))
                frame1 = put_Text(frame1, str(gender), (300, 120))
                frame1 = put_Text(frame1, str(beauty), (300, 190))
                frame1 = put_Text(frame1, "Age:", (50, 50))
                frame1 = put_Text(frame1, "Gender:", (50, 120))
                frame1 = put_Text(frame1, "Beauty:", (50, 190))
                #检测到人脸的图片，保存
                cv2.imwrite(str(time.time())+".jpg",frame1)
            else:
                frame1 = put_Text(frame1, "no face", (50, 50))
            display.clear_output(wait=True)
            img=frame1[:,:,::-1]
            plt.imshow(img)
            plt.axis('off') #不显示坐标
            plt.show()
            print("图片已经保存")
            servo.write_angle(int(beauty*2))
            led.write_digital(0)
            time.sleep(10)
        else:
            print("没有接摄像头或者摄像头被占用！")
        cap.release()
        cv2.destroyAllWindows()
    time.sleep(0.2)
```

In [ ]:

```

```