面向交叉学科的PYTHON程序设计与跨学科实践

PB19000352 易元昆

Level 1.15 基于 Python 的 pypdf2/pypdf4，对一个 PDF 文件进行文字内容提取

lab1

```
import PyPDF2 #调用pypdf2库

pdf = open('picture/test.pdf', 'rb') # 打开文件
pdfreader = PyPDF2.PdfFileReader(pdf)

# 读取内容
for i in range(0, pdfreader.getNumPages()):
    text = pdfreader.getPage(i).extractText()
    print(text)
    print('\n')
```

提取结果：

```
~/code/tencent_ubuntu/python  master !2 ?2  cd /home/ubuntu/code/tencent_ubuntu/python ;
 /usr/bin/env /bin/python3 /home/ubuntu/.vscode-server/extensions/ms-python.python-2022.4.1/pythonFile
s/lib/python/debugpy/launcher 36149 -- /home/ubuntu/code/tencent_ubuntu/python/ustc_python/lab1.py
 SEED Labs Œ Buffer Ov w V ulnerability Lab
 1
 Buf fer Ov w V ulnerability Lab
 Cop yright
 c

 2006 - 2014 W enliang Du, Syracuse Uni v ersity .
 The de v elopment of this document is/w as funded by three grants from the US National Science F ound
ation:
 A w ards No. 0231122 and 0618680 from TUES/CCLI and A w ard No. 1017771 from T rustw orth y Computing
.
 Permission is granted to cop y , distrib ute and/or modify this document under the terms of the GNU F
ree
 Documentation License, V ersion 1.2 or an y later v er sion published by the Free Softw are F oundati
on. A cop y
 of the license can be found at http://www .gnu.or g/licenses/fdl.html.
 1 Lab Ov er view
 The learning objecti v e of this lab is for students to g ain the e xperience on b uf fer -o v w vuln
er -
 ability by putting what the y ha v e learned about the vulnerability from class into action. Buf fer
o v w is
 as the condition in which a program attempts to write data be yond the boundaries of pre-allocated
 ed length b uf fers. This vulnerability can be utilized by a malicious user to alter the w control of
 the
 program, e v en e x ecute arbitrary pieces of code. This vulnerability arises due to the mixing of th
```

源于`picture/test.pdf`

Level 2.13对手机拍摄的任意照片进行 OCR 识别， 例如校园内的咖啡厅付款小票、车牌、路标、门牌等带有文字和数字的照片， 基于现有的 OCR库，对多类（至少三类）照片进行文字和数字提取

lab2

```python
from paddleocr import PaddleOCR

test = PaddleOCR(use_angle_cls=True, use_gpu=False)
result = test.ocr('ustc_python/source/lab2/5.jpg')
print(result)
```

```
/doc/fonts/simfang.ttf', warmup=False)
[2022/04/23 15:04:34] ppocr DEBUG: dt_boxes num : 18, elapse : 0.6466476917266846
[2022/04/23 15:04:34] ppocr DEBUG: cls num  : 18, elapse : 0.3101003170013428
[2022/04/23 15:04:35] ppocr DEBUG: rec_res num  : 18, elapse : 0.9683423042297363
[[[[1018.0, 196.0], [2088.0, 196.0], [2088.0, 432.0], [1018.0, 432.0]], ('#4饿了么', 0.97967464)], [[[
1046.0, 452.0], [2587.0, 442.0], [2587.0, 555.0], [1047.0, 565.0]], ('订单号: 8539618487883862460', 0.
9587986)], [[[1065.0, 599.0], [2702.0, 594.0], [2703.0, 712.0], [1066.0, 717.0]], ('下单时间: 2022/04/
23 11:00:23', 0.95488673)], [[[1035.0, 689.0], [1644.0, 709.0], [1634.0, 1043.0], [1024.0, 1023.0]], (
'姓名: 易林', 0.6821066)], [[[1075.0, 972.0], [2261.0, 972.0], [2261.0, 1213.0], [1075.0, 1213.0]], ('
电话: 18430777345.520', 0.9555345)], [[[1118.0, 1223.0], [2980.0, 1212.0], [2981.0, 1413.0], [1119.0,
1424.0]], ('地址: 中国科学技术大学中校区放北门', 0.98798585)], [[[1090.0, 1512.0], [1618.0, 1512.0], [
1618.0, 1831.0], [1090.0, 1831.0]], ('菜品名称', 0.7866149)], [[[2578.0, 1537.0], [2832.0, 1537.0], [2
832.0, 1787.0], [2578.0, 1787.0]], ('数量', 0.96219724)], [[[3490.0, 1546.0], [3744.0, 1546.0], [3744.
0, 1792.0], [3490.0, 1792.0]], ('实付', 0.99841297)], [[[1161.0, 1807.0], [2657.0, 1781.0], [2660.0, 1
968.0], [1164.0, 1994.0]], ('外送招牌特制牛肉饭（特盛', 0.9335279)], [[[3406.0, 1971.0], [3718.0, 1958
.0], [3727.0, 2179.0], [3415.0, 2192.0]], ('40.00', 0.9752388)], [[[3442.0, 2185.0], [3715.0, 2185.0],
 [3715.0, 2415.0], [3442.0, 2415.0]], ('2.00', 0.97950983)], [[[1131.0, 2205.0], [1835.0, 2193.0], [18
40.0, 2453.0], [1136.0, 2466.0]], ('（外送）打包盒', 0.9118617)], [[[3442.0, 2528.0], [3725.0, 2528.0]
, [3725.0, 2783.0], [3442.0, 2783.0]], ('0.50', 0.81933445)], [[[1123.0, 2563.0], [1493.0, 2563.0], [1
493.0, 2852.0], [1123.0, 2852.0]], ('配送费', 0.9939632)], [[[1114.0, 3088.0], [1363.0, 3088.0], [1363
.0, 3387.0], [1114.0, 3387.0]], ('优惠', 0.99836636)], [[[3581.0, 3314.0], [3725.0, 3314.0], [3725.0,
3417.0], [3581.0, 3417.0]], ('00', 0.5058401)]]
```

基于图片`5.jpg`即外卖小票的识别结果

```
=False, use_onnx=False, use_pdserving=False, use_space_char=True, use_tensorrt=False, vis_font_path='.
/doc/fonts/simfang.ttf', warmup=False)
[2022/04/23 15:08:53] ppocr DEBUG: dt_boxes num : 14, elapse : 0.6220865249633789
[2022/04/23 15:08:53] ppocr DEBUG: cls num  : 14, elapse : 0.1633000373840332
[2022/04/23 15:08:55] ppocr DEBUG: rec_res num  : 14, elapse : 1.4350509643554688
[[[[1174.0, 1038.0], [1638.0, 1038.0], [1638.0, 1092.0], [1174.0, 1092.0]], ('（上市许可持有人）', 0.9
0087914)], [[[1167.0, 1088.0], [2124.0, 1092.0], [2124.0, 1158.0], [1167.0, 1154.0]], ('名称：石家庄以
岭药业股份有限公司', 0.9440371)], [[[1376.0, 1158.0], [2394.0, 1158.0], [2394.0, 1213.0], [1376.0, 121
3.0]], ('家庄市高新技术开发区天山大街238号', 0.97571415)], [[[1170.0, 1216.0], [1303.0, 1216.0], [1303
.0, 1271.0], [1170.0, 1271.0]], ('生产', 0.83523434)], [[[1306.0, 1221.0], [1453.0, 1212.0], [1456.0,
1258.0], [1309.0, 1268.0]], ('企业', 0.9998858)], [[[1159.0, 1271.0], [2253.0, 1275.0], [2253.0, 1341.
0], [1159.0, 1337.0]], ('企业名称：石家庄以岭药业股份有限公司', 0.98296857)], [[[1159.0, 1325.0], [304
4.0, 1329.0], [3044.0, 1407.0], [1159.0, 1403.0]], ('生产地址：石家庄市高新技术并发区天山大街238号 邮
政编码：050035', 0.91299605)], [[[1156.0, 1391.0], [2109.0, 1399.0], [2109.0, 1465.0], [1155.0, 1457.0
]], ('电话号码：8008038581（座机拨打）', 0.97695905)], [[[2098.0, 1407.0], [3051.0, 1407.0], [3051.0,
1461.0], [2098.0, 1461.0]], ('4007898989（手机/座机均可拨打）', 0.92472583)], [[[1448.0, 1457.0], [191
5.0, 1461.0], [1915.0, 1520.0], [1448.0, 1516.0]], ('(0311)85901719', 0.94517523)], [[[1151.0, 1523.0]
, [1919.0, 1523.0], [1919.0, 1590.0], [1151.0, 1590.0]], ('传真号码：（0311)85901719', 0.9549655)], [[
[1323.0, 1582.0], [2269.0, 1586.0], [2268.0, 1652.0], [1322.0, 1648.0]], ('址：http://www.yilingshop.c
om', 0.9510314)], [[[1151.0, 1593.0], [1208.0, 1593.0], [1208.0, 1648.0], [1151.0, 1648.0]], ('网', 0.
9983712)]]
```

基于图片`6.jpg`即商标标识的识别结果。

```
/doc/fonts/simfang.ttf', warmup=False)
[2022/04/23 15:10:20] ppocr DEBUG: dt_boxes num : 1, elapse : 0.6778135299682617
[2022/04/23 15:10:20] ppocr DEBUG: cls num  : 1, elapse : 0.04416179656982422
[2022/04/23 15:10:20] ppocr DEBUG: rec_res num  : 1, elapse : 0.07888960838317871
[[[[846.0, 892.0], [2664.0, 869.0], [2669.0, 1316.0], [851.0, 1338.0]], ('火警119', 0.937353)]]
```

基于图片`7.jpg`即火警标识的识别结果。

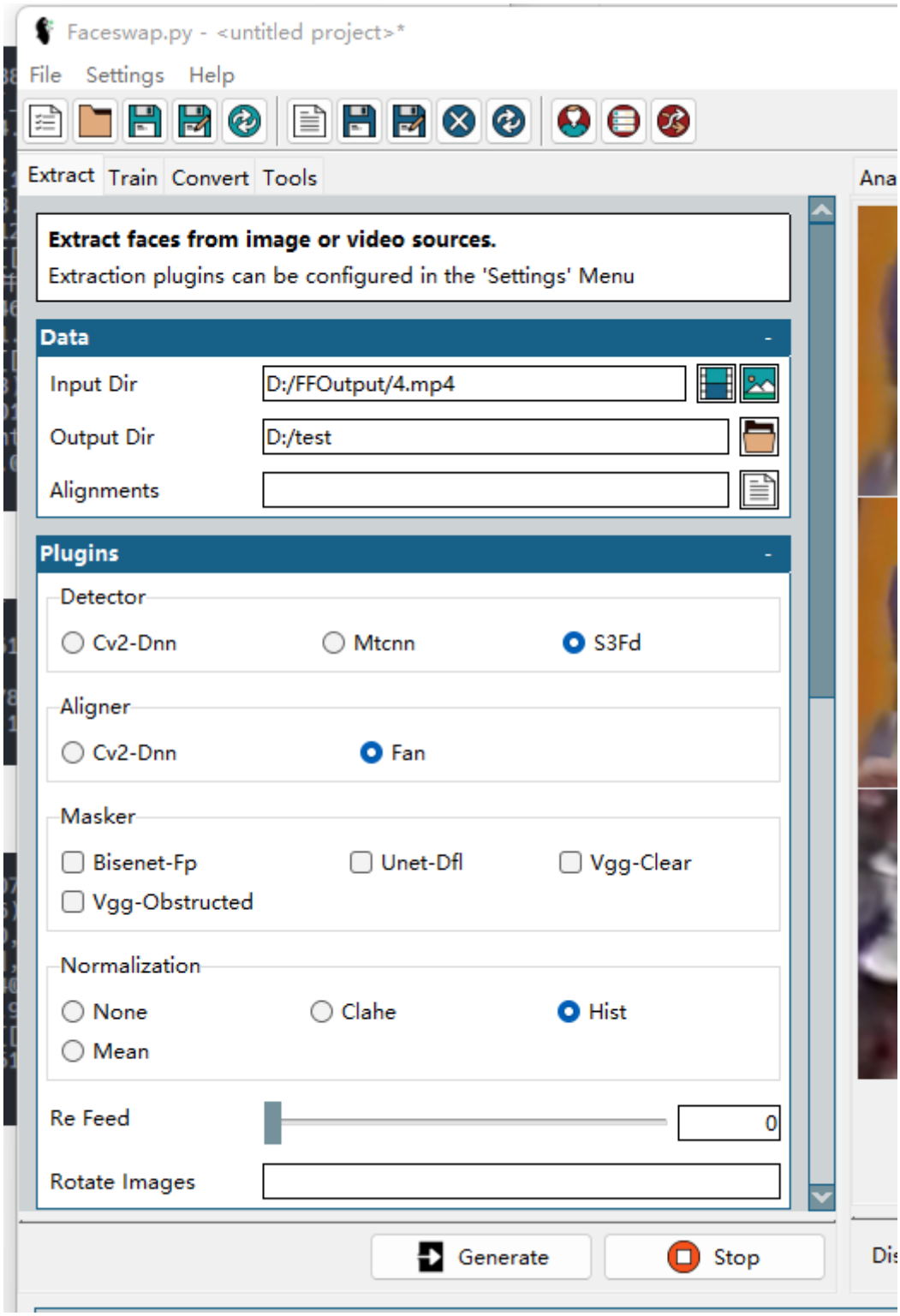
```
[2022/04/23 15:11:18] ppocr DEBUG: cls num  : 9, elapse : 0.13250851631164455
[2022/04/23 15:11:19] ppocr DEBUG: rec_res num  : 9, elapse : 0.49456334114074707
[[[[171.0, 2.0], [209.0, 2.0], [209.0, 18.0], [171.0, 18.0]], ('64%', 0.92005616)], [[[3.0, 3.0], [40.
0, 3.0], [40.0, 19.0], [3.0, 19.0]], ('GPU1', 0.8505967)], [[[90.0, 3.0], [114.0, 3.0], [114.0, 18.0],
 [90.0, 18.0]], ('80', 0.97263336)], [[[235.0, 2.0], [374.0, 0.0], [375.0, 19.0], [235.0, 21.0]], ('18
45WHz71.2V', 0.8447757)], [[[63.0, 21.0], [133.0, 21.0], [133.0, 40.0], [63.0, 40.0]], ('C1605"', 0.72
837806)], [[[90.0, 44.0], [115.0, 44.0], [115.0, 62.0], [90.0, 62.0]], ('66', 0.9929782)], [[[233.0, 4
2.0], [301.0, 42.0], [301.0, 62.0], [233.0, 62.0]], ('3788以z', 0.75518185)], [[[170.0, 45.0], [196.0,
 45.0], [196.0, 59.0], [170.0, 59.0]], ('54', 0.99775827)], [[[316.0, 44.0], [351.0, 44.0], [351.0, 61
.0], [316.0, 61.0]], ('54.', 0.7999828)]]
```

基于图片`4.png`游戏帧率截屏的识别结果。

level3 faceswap换脸

首先安装faceswap及有关环境，然后通过`python faceswap.py gui`运行faceswap图形环境。

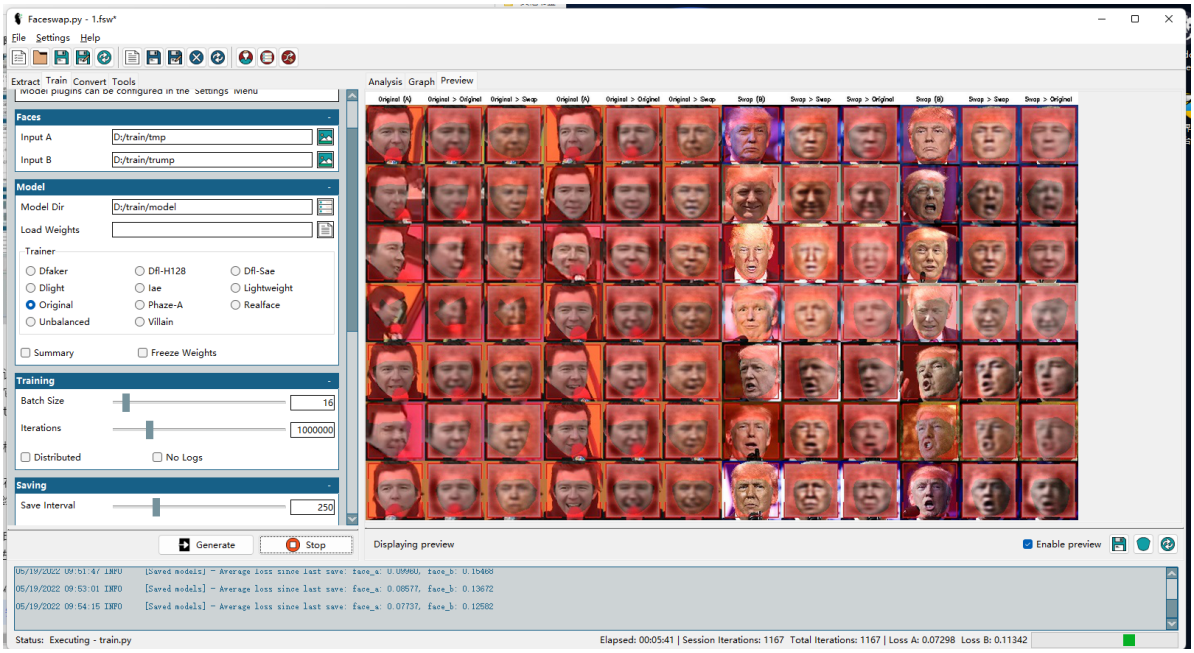
> faceswap需要tensorflow环境，最好拥有Nvidia GPU用于加速训练，因此本机安装了用于深度学习的tensorflow-gpu，及相应的CUDA环境，环境为CUDA-v11.5，配套的cuDNn为v8.22。



然后首先导入训练用的视频用于换脸任务的训练。训练视频时长约为1.30min，视频内容为`Rick roll`

然后再导入需要替换的人脸信息`Trump`(共使用41张图片)

然后进行训练：



训练2h后停止训练，然后通过`Convert`生成换脸后视频

原视频为位于`lab3/origin.mp4`，生成的视频为`lab3/origin_converted.py`

通过观察换脸后视频可以得知，需要不断调整参数使得效果最好。而且当视频中人脸过小时换脸效果较差，faceswap更加适合对人脸占比较大的视频进行换脸。

Level 4 face_recognition

通过`conda`创建虚拟环境，然后安装`face_recognition`所需资源

`cmake，Visual Studio for C++`已经安装

只需要通过`pip install dlib face_recognition`安装就能够完成构建

获取图片

find_face

定位图片中人脸的位置：

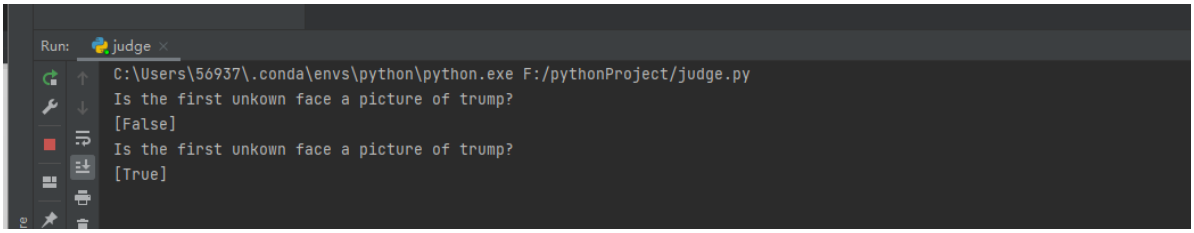
> 可以选择通过CNN神经网络或者通过普通方式进行

提取脸部特征点

find_features

判断人脸是否为同一张人脸

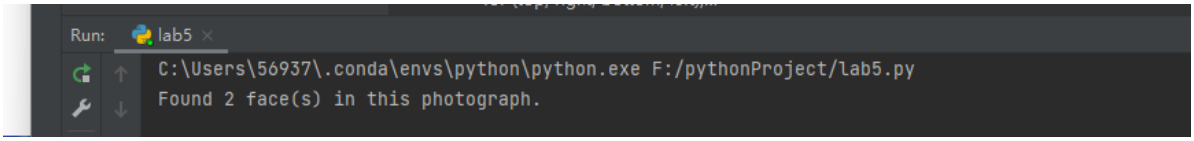
judge



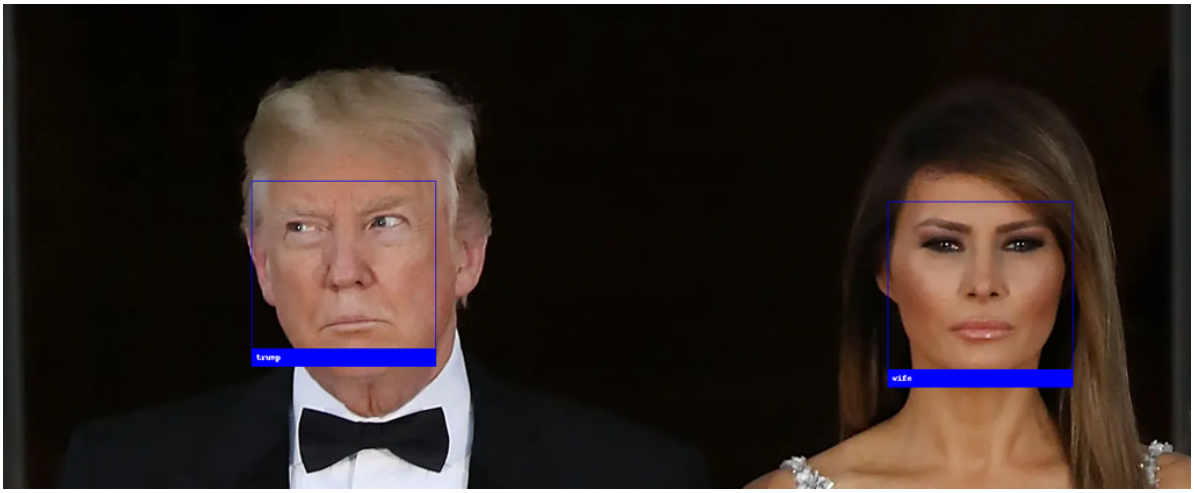
Level 5 自定义人脸识别

通过对`face_recognition`代码进行组合，可以进行图片人脸数量的确定和人脸数据的匹配

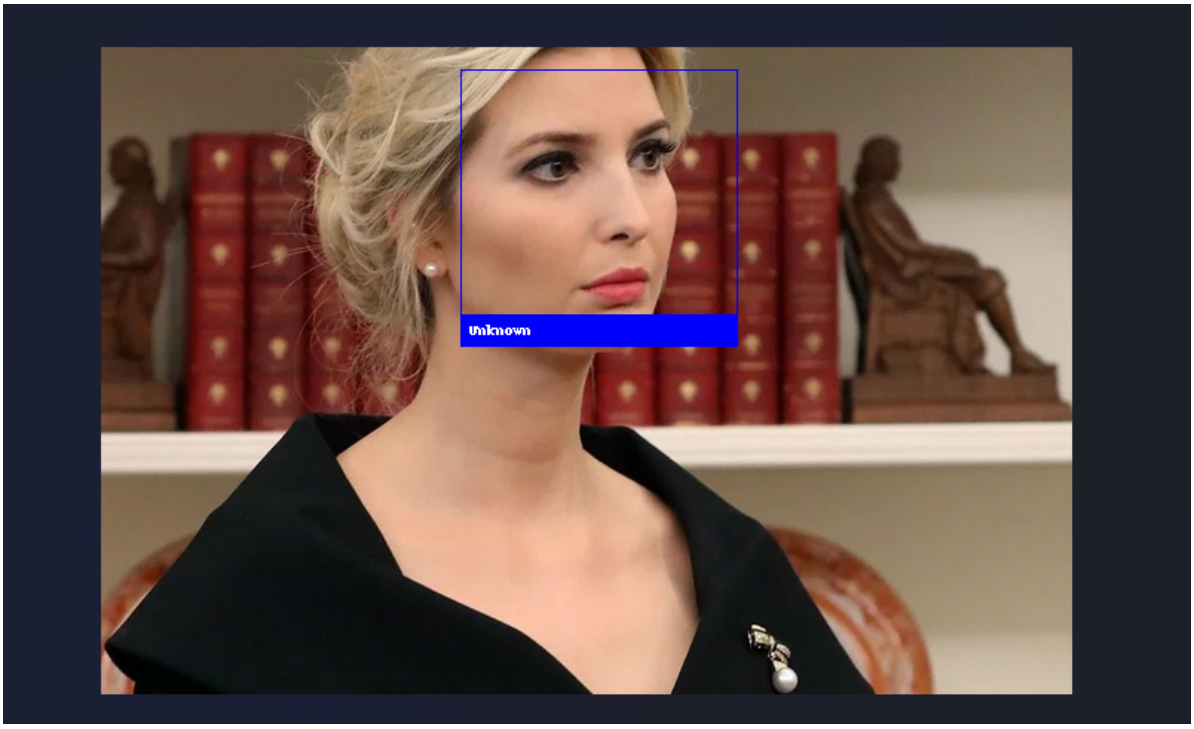
- 确定人脸数量



- 识别人脸并进行姓名标注



如果是不在数据库中的人脸则显示：



总结与收获

进一步了解了python的使用与库的调用操作。