

# OpenCV Python + Tesseract-OCR輕鬆實現中文識別

原創 陰沉魚 OpenCV學堂 昨天



點擊上方藍字關注我們

微信公眾號：[OpenCV學堂](#)

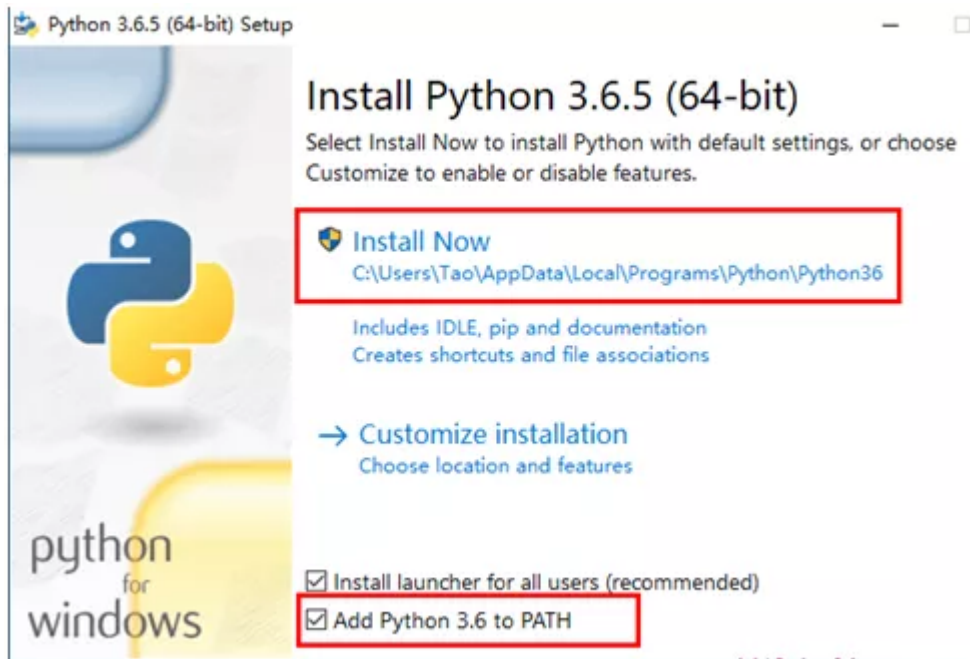
關注獲取更多計算機視覺與深度學習知識

## 01 軟件版本

- Python3.6.5
- OpenCV-Python 4.x
- Tesseract-OCR 5.0.0-alpha.20201127
- Win10 64

## 02 安裝與配置

安裝Python SDK選擇默認安裝，同時勾選最下面得紅色矩形框內容。



安裝opencv-python開發包

```
1 pip install opencv-python
```

安裝Tesseract-OCR Python SDK支持

```
1 pip install pytesseract
```

下載Tesseract-OCR 5.0.0-alpha.20201127安裝包並安裝！然後在環境變量中添加

```
1 C:\Program Files\Tesseract-OCR
```

### 03 驗證與測試

安裝與配置好OpenCV-Python與Tesseract-OCR之後，需要進一步通過代碼驗證正確性。打開Pycharm IDE，新建一個python項目與python文件，輸入以下代碼

```
1 import pytesseract as tess
2 print(tess.get_tesseract_version())
3 print(tess.get_languages())
```

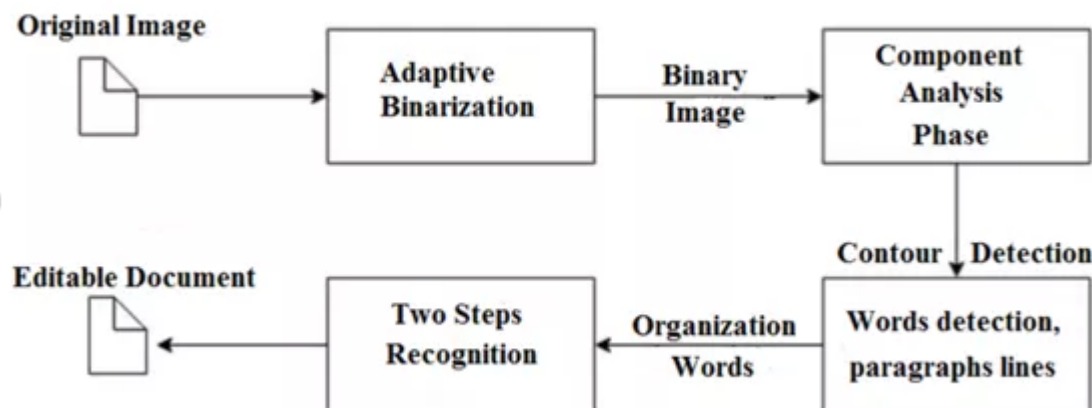
運行結果如下：

```
5.0.0-alpha.20201127
['eng', 'osd']
```

第一行是版本信息，第二行是支持的語言信息，默認只支持英文。

### Tesseract-OCR介紹

開源的OCR識別引擎，高版本識別基於LSTM，其整個處理流程如下：



檢測與識別文本函數：

檢測：

```
1 def image_to_boxes(  
2     image,  
3     lang=None,  
4     config='',  
5     nice=0,  
6     output_type=Output.STRING,  
7     timeout=0,  
8 )
```

返回所有識別文字的Box框坐標，每一行為一個BOX信息輸出

每行的前五個值分別是，識別的字符、BOX框的左上角與右下角坐標

## 識別

```
1 def image_to_string(  
2     image,  
3     lang=None,  
4     config='',  
5     nice=0,  
6     output_type=Output.STRING,  
7     timeout=0,  
8 )
```

輸入的圖像通道順序是RGB，OpenCV默認為BGR，返回的是識別結果

必輸入的參數是image，其它可選

英文与数字识别

Tesseract-OCR默认支持英文与数字识别，有输入图像如下：



Python代码如下：

```
image = cv.imread("D:/cv4j.png")
image_rgb = cv.cvtColor(image, cv.COLOR_BGR2RGB)
text = tess.image_to_string(image_rgb, lang="eng")
content = text.replace("\f", "").split("\n")
for c in content:
    if len(c) > 0:
        print(c)
h, w, c = image.shape
boxes = tess.image_to_boxes(image)
for b in boxes.splitlines():
    b = b.split(' ')
    image = cv.rectangle(image, (int(b[1]), h - int(b[2])), (int(b[3]), h - int(b[4]

cv.imshow('text detect', image)
cv.waitKey(0)
cv.destroyAllWindows()
```

运行结果如下：



英文识别对无干扰的背景效果非常好！

### 中文识别

默认情况下Tesseract-OCR不支持中文识别，需要下载中文识别的模型文件，然后放置到安装路径的tessdata目录下：

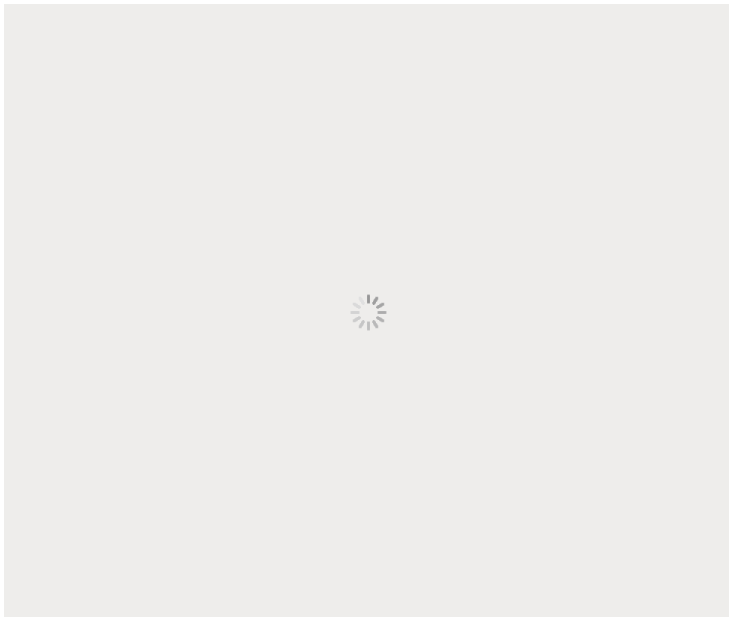
```
1 C:\Program Files\Tesseract-OCR\tessdata
```

然后在运行语言检查支持代码，运行结果如下：



其中chi\_sim表示中文简体支持，eng表示英文支持！

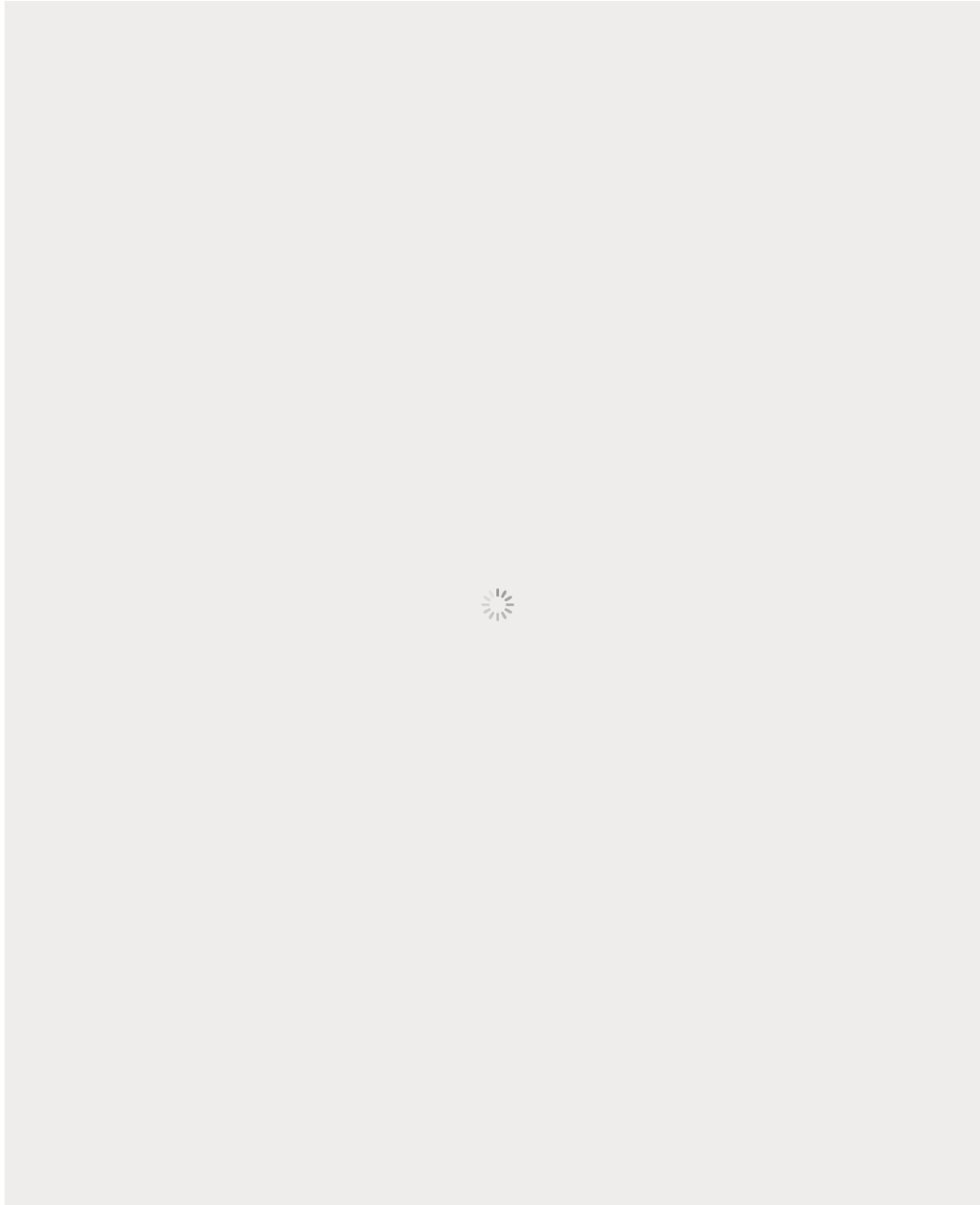
**以下图为例：**



## 识别代码

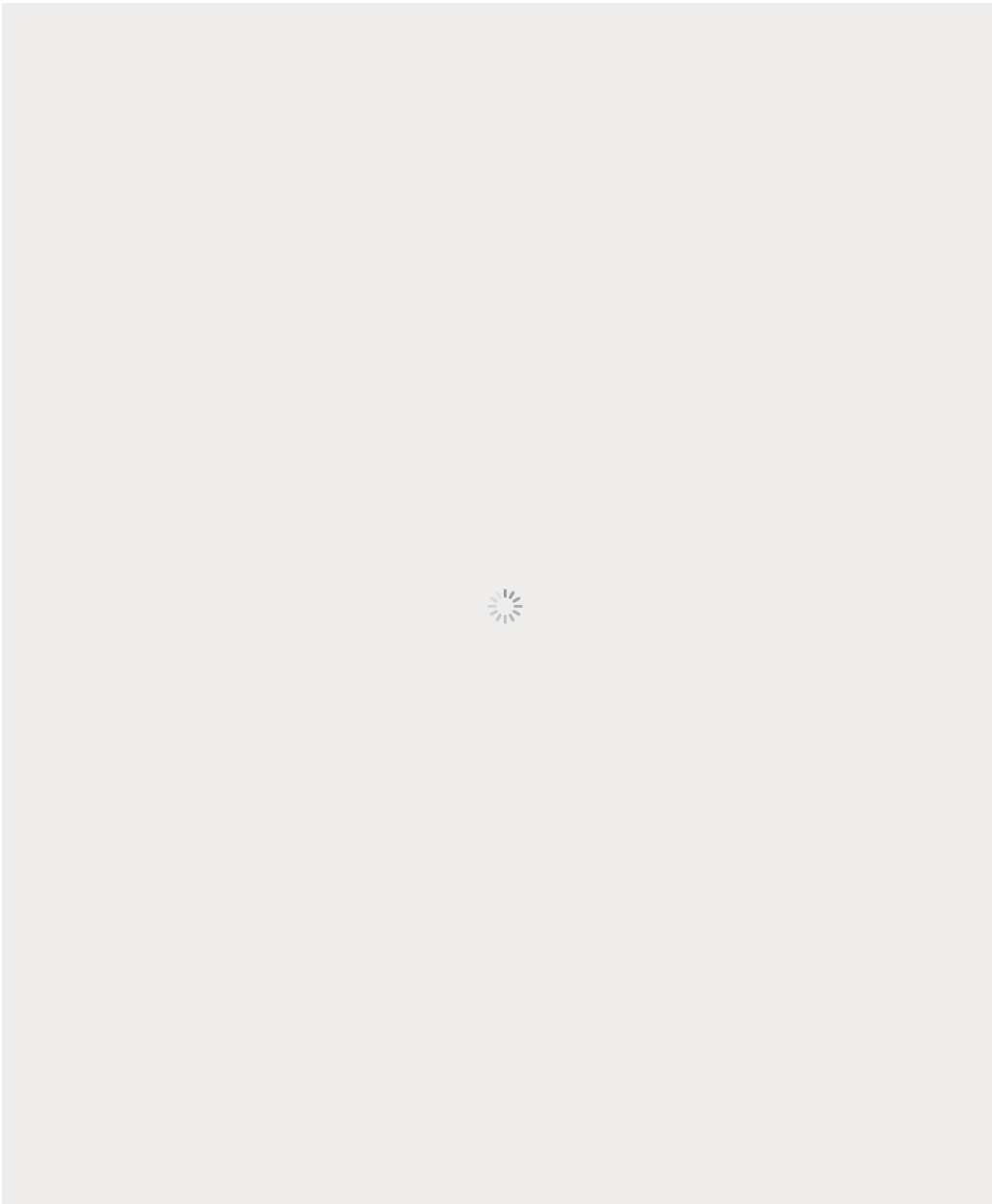
```
image = cv.imread("D:/yanxishe.png")
image_rgb = cv.cvtColor(image, cv.COLOR_BGR2RGB)
text = tess.image_to_string(image_rgb, lang="chi_sim")
print(text)
h, w, c = image.shape
boxes = tess.image_to_boxes(image)
for b in boxes.splitlines():
    b = b.split(' ')
    image = cv.rectangle(image, (int(b[1]), h - int(b[2])), (int(b[3]), h - int(b[4]
cv.imshow('text detect', image)
cv.waitKey(0)
cv.destroyAllWindows()
```

## 运行结果



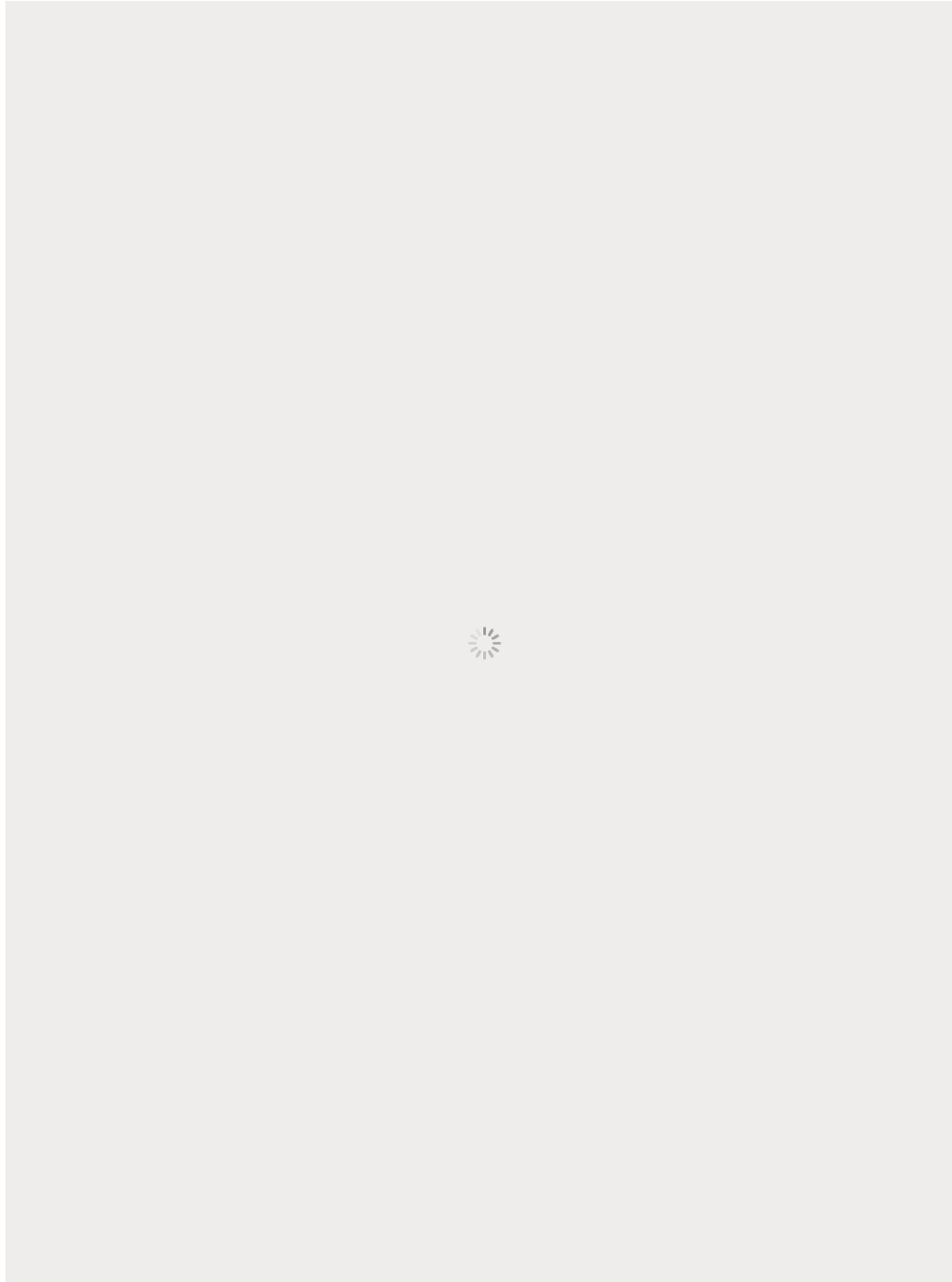
正确率还不错，需要进一步处理一下，直接放大两倍，然后再测试一波





发现错误识别大大减少，基本可用！

**总结一下：**使用OpenCV预处理可以大大降低Tesseract-OCR的误识别率！



直接识别原图，发现图像上的灰度文字被忽视，无法识别！

## 君子藏器于身，待时而动

### 推荐阅读

[OpenCV4系统化学习路线图-视频版本！](#)

[Tensorflow + OpenCV4 安全帽检测模型训练与推理](#)

[汇总 | OpenCV DNN模块中支持的分类网络](#)

[OpenCV中支持的人脸检测方法整理与汇总](#)

[从3D人脸到自动驾驶，CVPR2020十个顶级开源数据集](#)

[轻松学Pytorch – 构建UNet实现道路裂纹检测](#)

[详解ENet | CPU可以实时的道路分割网络](#)

[从Pytorch 的ONNX到OpenVINO中IR中间层](#)

[OpenVINO中的FCOS人脸检测模型代码演示](#)

[详解OpenVINO 模型库中的人脸检测模型](#)

[效果特好 | OpenVINO手写数字识别](#)

[OpenVINO+OpenCV 文本检测与识别](#)

[OpenVINO + OpenCV实现点头与摇头识别验证](#)

[OpenVINO + OpenCV实现车辆检测与道路分割](#)

[OpenCV二值图像分析之寻找缺失与靶心](#)

[OpenCV 基于Inception模型图像分类](#)

[OpenCV4.4 + YOLOv4 真的可以运行了.....](#)

[告别对象检测图像数据手动标注的坑，用Anno-Mage半自动图像标注工具](#)

[经验 | OpenCV图像旋转的原理与技巧](#)



喜歡此內容的人還喜歡

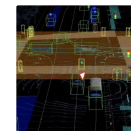
你真的理解圖像處理經典算法SIFT 嗎？最深入、最全面綜述：尺度不變特徵轉換

極市平台



激光雷達+GPS+IMU+輪速計的傳感器融合的定位系統

計算機視覺life



Halcon缺陷檢測實例轉OpenCV實現(六) 瓶口缺陷檢測



