**博客链接：**[**http://www.cnblogs.com/yinzx/p/4741986.html**](http://www.cnblogs.com/yinzx/p/4741986.html)

**大概介绍下相关模块的概念**：

Python-tesseract 是光学字符识别Tesseract OCR引擎的Python封装类。能够读取任何常规的图片文件(JPG, GIF ,PNG , TIFF等)并解码成可读的语言。在OCR处理期间不会创建任何临文件

PIL （Python Imaging Library）是 Python 中最常用的图像处理库，目前版本为 1.1.7，我们可以 [在这里](http://www.pythonware.com/products/pil/index.htm) 下载学习和查找资料。

Image 类是 PIL 库中一个非常重要的类，通过这个类来创建实例可以有直接载入图像文件，读取处理过的图像和通过抓取的方法得到的图像这三种方法。

python对图像的处理比较常见的是用pytesseract识别验证码，要安装pytesseract库，必须先安装其依赖的PIL及tesseract-ocr，其中PIL为图像处理库，而后面的tesseract-ocr则为google的ocr识别引擎。**下载链接：http://www.waitalone.cn/python-php-ocr.html   该链接文档描述了如何配置相关环境，以及识别验证码的python代码，总结起来就三步：安装PIL.exe；安装**[tesseract-ocr-setup](http://tesseract-ocr.googlecode.com/files/tesseract-ocr-setup-3.02.02.exe)**.exe;安装pip install pytesseract**

1、识别图片简单的示例

[复制代码](javascript:void(0);)

1 import Image

2 import pytesseract

3

4 image=Image.open("1.jpg") #打开验证码图片

5 image.load() #加载一下图片，防止报错，此处可省略  
6 image.show() 　　　　#调用show来展示图片，调试用，可省略

7 vcode=pytesseract.image\_to\_string(image)

8 print vcode

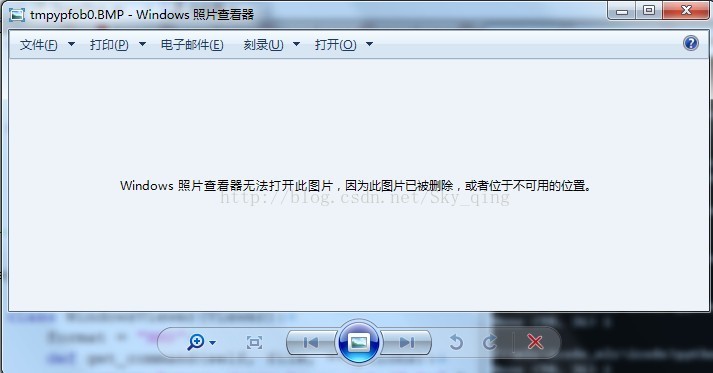
[复制代码](javascript:void(0);)

我按照该代码示例实际操作时报了2个错误：

1、The \_imaging C module is not installed

是因为环境配置的问题，我的系统是win7 64bit，开始的时候只安装了[PIL-1.1.7.win-amd64-py2.7.exe](https://github.com/lightkeeper/lswindows-lib/blob/master/amd64/python/PIL-1.1.7.win-amd64-py2.7.exe?raw=true)，把[PIL-1.1.7.win32-py2.7.exe](http://effbot.org/downloads/PIL-1.1.7.win32-py2.7.exe)安装后，在python命令行输入：import \_imaging，如果不报错则问题应该就解决了

2、在 win 7 下使用 PIL 中的 Image 模块的 show() 函数时，报如下错：



解决方法：将 PIL 安装目录下的 ImageShow.py 文件的第 99 行：（我的 python 安装在 C:\python 2.7， ImageShow.py  文件在：C:\Python27\Lib\site-packages\PIL\ImageShow.py）

return "start /wait %s && del /f %s" % (file, file)

替换为：

return "start /wait %s && PING 127.0.0.1 -n 5 > NUL && del /f %s" % (file, file)

2、实际情况中，电脑本地并没有验证码图片，都是从网页中的验证码图片链接抓取的，实际示例：

[复制代码](javascript:void(0);)

1 import urllib2

2 import pytesseract

3 import Image

4 import time

5

6 fp=urllib2.urlopen('https://vcs.suning.com/vcs/imageCode.htm?uuid=1e68d06a-1134-410b-9606-f0eb4ae23bbe')　　#示例网址的验证码图片地址

7 print fp

8

9 name="D:\\curl\\1.jpg"

10 f=open(name,'wb')

11 f.write(fp.read())　　　　　　#读取验证码图片并保存为1.jpg

12 time.sleep(2)

13 f.close()

14

15 fl=open(name,'rb')

16 image=Image.open(fl)

17 image.show()

18 #image.load()

19 vcode=pytesseract.image\_to\_string(image)

20 print vcode

21 fl.close()

[复制代码](javascript:void(0);)

以上本亲测可行，不过验证码识别率不是很高，简单的图片识别率还是可以的

**附录：Image模块用法介绍【转】**

1. 简介。

    图像处理是一门应用非常广的技术，而拥有非常丰富第三方扩展库的 Python 当然不会错过这一门盛宴。PIL （Python Imaging Library）是 Python 中最常用的图像处理库，目前版本为 1.1.7，我们可以 [在这里](http://www.pythonware.com/products/pil/index.htm) 下载学习和查找资料。

    Image 类是 PIL 库中一个非常重要的类，通过这个类来创建实例可以有直接载入图像文件，读取处理过的图像和通过抓取的方法得到的图像这三种方法。

2. 使用。

    导入 Image 模块。然后通过 Image 类中的 open 方法即可载入一个图像文件。如果载入文件失败，则会引起一个 IOError ；若无返回错误，则 open 函数返回一个 Image 对象。现在，我们可以通过一些对象属性来检查文件内容，即：

1 >>> import Image  
2  >>> im = Image.open("j.jpg")  
3  >>> print im.format, im.size, im.mode  
4 JPEG (440, 330) RGB

    这里有三个属性，我们逐一了解。

        format : 识别图像的源格式，如果该文件不是从文件中读取的，则被置为 None 值。

        size : 返回的一个元组，有两个元素，其值为象素意义上的宽和高。

        mode : RGB（true color image），此外还有，L（luminance），CMTK（pre-press image）。

    现在，我们可以使用一些在 Image 类中定义的方法来操作已读取的图像实例。比如，显示最新载入的图像：

1 >>>im.show()  
2  >>>