

Python 数字图像处理–scikit–image 库学习开始

2018 年 11 月 9 日 BY [HARLEY](#) · 0 评论

常用 Python 数字处理图像库

基于 python 脚本语言开发的数字图片处理库，比如 PIL,Pillow, opencv, scikit–image 等。(PIL 是针对 python2,pillow 是针对 python3, 两者功能一样。)

PIL 和 Pillow 只提供最基础的数字图像处理，功能有限；opencv 实际上是一个 c++ 库，只是提供了 python 接口，更新速度非常慢。scikit–image 是基于 scipy 的一款图像处理库，它将图片表示为数组 (NumPy arrays) 进行处理，正好与 matlab 一样。

Image 读出来的是 PIL 的类型。而 skimage.io 读出来的数据是 numpy 格式的，这方便我们与 TensorFlow 框架对接。因此，我最终选择 scikit–image 进行数字图像处理。

注意本文只是简单的介绍了下 skimage 库的基本用法，包括子模块 data 和 io 加载图像并获取图像信息，更多信息请参考官网教程。

scikit–image 库了解

skimage 包的全称是 scikit–image SciKit (toolkit for SciPy) ，它对 scipy.ndimage 进行了扩展，提供了更多的图片处理功能。它是由 python 语言编写的，由 scipy 社区开发和维护。

模块导入

scikit–image 是一个将图像作为 numpy array 格式处理的 Python 包。该包被导入为 skimage：

```
>>> import skimage
```

skimage 库子模块和函数列表

这个[网页](#)提供了子模块和函数列表。skimage 包由许多的子模块组成，各个子模块提供不同的功能。主要子模块列表如下：

子模块名称	主要实现功能
io	读取、保存和显示图片或视频
data	提供一些测试图片和样本数据
color	颜色空间变换
filters	图像增强、边缘检测、排序滤波器、自动阈值等
draw	操作于 numpy 数组上的基本图形绘制，包括线条、矩形、圆和文本等
transform	几何变换或其它变换，如旋转、拉伸和拉东变换等
morphology	形态学操作，如开闭运算、骨架提取等
exposure	图片强度调整，如亮度调整、直方图均衡等
feature	特征检测与提取等
measure	图像属性的测量，如相似性或等高线等

segmentation	图像分割
restoration	图像恢复
util	通用函数

skimage 初步学习

在 skimage 子模块中，我们可以找到绝大多数数字图像处理的功能。对于读取外部图片并显示，skimage 提供了 io 子模块，顾名思义，这个模块是用来图片输入输出操作的。为了方便练习，skimage 提供 data 子模块，里面嵌套了一些示例图片，我们可以直接使用。

下面是 data 子模块的用法：

```
>>> from skimage import data    #导入 data 子模块

>>> camera = data.camera()      #create camera image,return
(512, 512) uint8 ndarray
```

一要注意，在 scikit-image 中，图像表示为 NumPy 阵列，例如用于灰度图像的为 2-D（2 维）阵列。代码示例：

```
1 import skimage
2 from skimage import data
3 #create camera image,return (512, 512) uint8 ndarray
4 camera = data.camera()
5 print(type(camera))    #显示图像的数据类型
6 print(camera.shape)    #显示图像的尺寸
```

```
<class 'numpy.ndarray'>
(512, 512)
```

```
>> import skimage

>> from skimage import data    #create camera image,return (300,
451, 3) uint8 ndarray

>> camera = data.camera()
```

```
>> print(type(camera))      #显示图像的数据类型
>>> print(camera.shape)     #显示图像的尺寸
```

代码解释: skimage.data 子模块提供了一组返回示例图像的函数, 可用于快速操作图像文件。

也可以使用 skimage.io.imread () 从外部文件中读取图像文件, 并作为 Numpy 数组加载。

```
1 import os
2 filename = os.path.join(skimage.data_dir, 'moon.png')
3 from skimage import io
4 moon = io.imread(filename)
5 print('The image moon data\n',moon)

The image moon data
[[116 116 122 ...,  93  96  96]
 [116 116 122 ...,  93  96  96]
 [116 116 122 ...,  93  96  96]
 ...,
 [109 109 112 ..., 117 116 116]
 [114 114 113 ..., 118 118 118]
 [114 114 113 ..., 118 118 118]]
```

```
>>> import os
>>> filename = os.path.join(skimage.data_dir, 'moon.png')
>>> from skimage import io
>>> moon = io.imread(filename)
>>> print('The image moon data\n',moon)
```

参考链接

http://scikit-image.org/docs/stable/user_guide/getting_started.html