

华东师范大学软件工程实验报告

谢嘉东 10185101247

陈俊潼 10185101210

March 2020

- 课程名称：数字图像处理
- 年级：2018 级本科
- 实验编号：实验 001
- 上机实践日期：2020.3

目录

1 分工情况	2
2 实验内容	2
3 实验目的	2
4 实验原理	2
5 实验方法	3
5.1 读取和显示图像	3
5.2 改变图片形式	3
5.3 改变图片大小与旋转图片	4
6 实验结果及分析	4
7 主要核心代码	8
7.1 读取与显示图像	8
7.2 改变图像形式	8
7.3 改变图片大小与旋转图片	9
8 参考资料	10

1 分工情况

第一次实验相对比较容易，故我们组内两人均完成了实验部分。

实验报告由两人共同撰写。

2 实验内容

本实验主要为掌握 PIL 库和 Python 的基本使用方法，含以下内容：

- 1 将 JPG 格式图像文件进行读取和显示。
- 2 将 RGB 彩色图像转变为灰度图像并保存。
- 3 将原图转化为“二值图像”并保存。
- 4 将所给的图片尺寸大小分别调整为 160×120 、 320×240 、 640×480 ，并对原图进行 90 度和 180 度的旋转变化，将变换后的每个不同结果都保存为 .PNG 格式。

3 实验目的

- 1 掌握 PIL 库的基本使用方法
- 2 熟悉 Python 的基本语句和文件相关操作
- 3 理解 RGB 图像在计算机中的存储格式
- 4 理解二值图像、灰度等基本概念
- 5 熟悉 JPG 、 PNG 格式图像的存储方式以及读取方式

4 实验原理

利用 Python 库 Pillow 中的 PIL 包提供的函数 `resize()`, `point()`, `thumbnail()`, `convert()` 等实现各种处理效果，函数的使用方法是：

- `Image.resize(size, resample=0)`
 - size –The requested size in pixels, as a 2-tuple: (width, height).
 - resample –An optional resampling filter.
- `Image.point(lut, mode=None)`
 - lut –A lookup table, containing 256 (or 65336 if self.mode==”I” and mode == “L”) values per band in the image.
 - mode –Output mode (default is same as input).
- `Image.thumbnail(size, resample=3)`
 - size –Requested size.
 - resample –Optional resampling filter.

- `Image.convert(mode=None, matrix=None, dither=None, palette=0, colors=256)`
 - mode –The requested mode.
 - matrix –An optional conversion matrix. If given, this should be 4- or 12-tuple containing floating point values.
 - dither –Dithering method, used when converting from mode “RGB” to “P” or from “RGB” or “L” to “1”. Available methods are NONE or FLOYDSTEINBERG (default).
 - palette –Palette to use when converting from mode “RGB” to “P” . Available palettes are WEB or ADAPTIVE.
 - colors –Number of colors to use for the ADAPTIVE palette. Defaults to 256.

5 实验方法

5.1 读取和显示图像

使用 `Image.open()` 函数之后再使用 `im.show()` 即可调用系统默认的图像浏览程序打开图片。

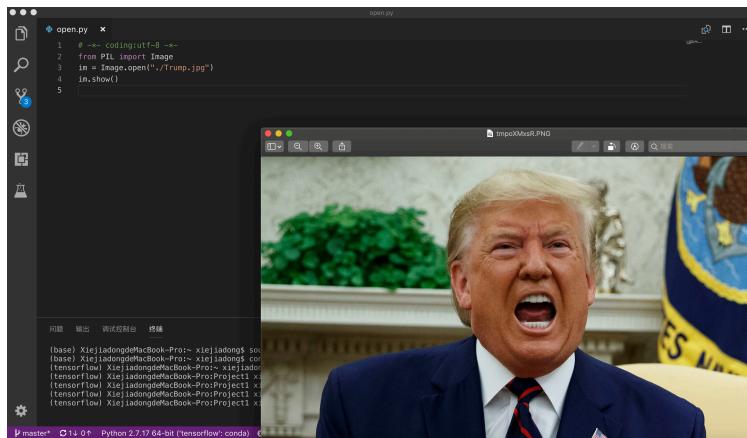


图 1: 读取和显示效果图

5.2 改变图片形式

转成灰度图像 使用 `im.convert('L')` 语句即可转成灰度图像。

转成二值图像 首先将图像转换成灰度图像，再新建一个数组作为映射表。如果色彩深度大于阈值则存 1，小于阈值则存 0。接着再使用 `im.point()` 函数，将映射表传递过去并设置图像模式为二值图像，PIL 库就会逐个遍历像素，将灰度图像的灰度值在数组中映射成 0 或 1，最后得到的图像便是二值图像。

实验中我们尝试了不同阀值所带来的效果，最终将灰度值阈值设置为 110，相对而言可以取得较好效果。

5.3 改变图片大小与旋转图片

改变图片大小 使用 `thumbnail()` 函数即可改变图片大小。

旋转图片 如果是规则的 90、180、270 度旋转可以直接使用 PIL 库的 `transpose()` 函数实现。

之后我们尝试在这一块进行了拓展探究，我们尝试将图片旋转 45 度，但是发现，直接使用 `rotate()` 函数进行旋转操作，会导致的图片裁剪。



图 2: 被裁减的效果

于是，我们探究出了解决这个问题的办法：先用三角函数计算出旋转后并填充黑边的图片大小，新建黑画布，再将原图居中复制到新的画布中，这时使用 `rotate()` 就不会将图片裁剪了，结果便会在实验结果中呈现。

6 实验结果及分析

基本完成了实验预期所要达到的要求，并在图片旋转方面，做了一定的拓展实践。
最终的实验结果如下：

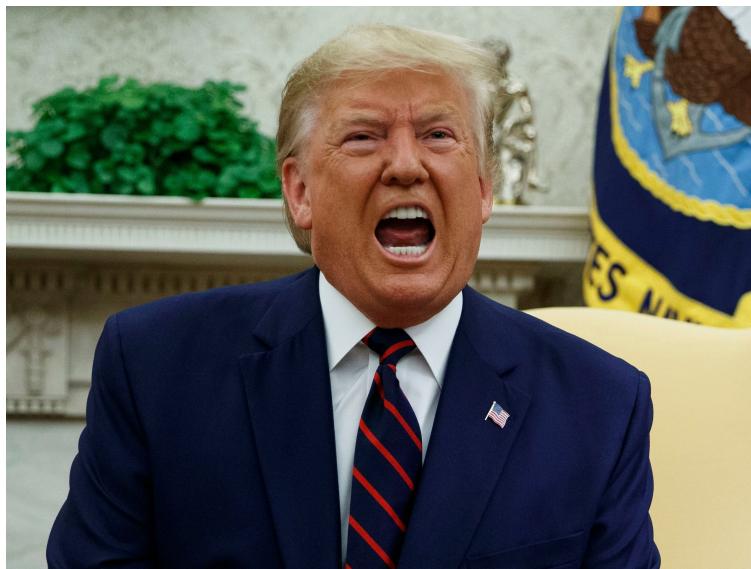


图 3: 原图



图 4: 灰度图像



图 5: 二值图像 (灰度阈值 = 110)



(a) 160 * 120

(b) 320 * 240

(c) 640 * 480

图 6: 缩放图像

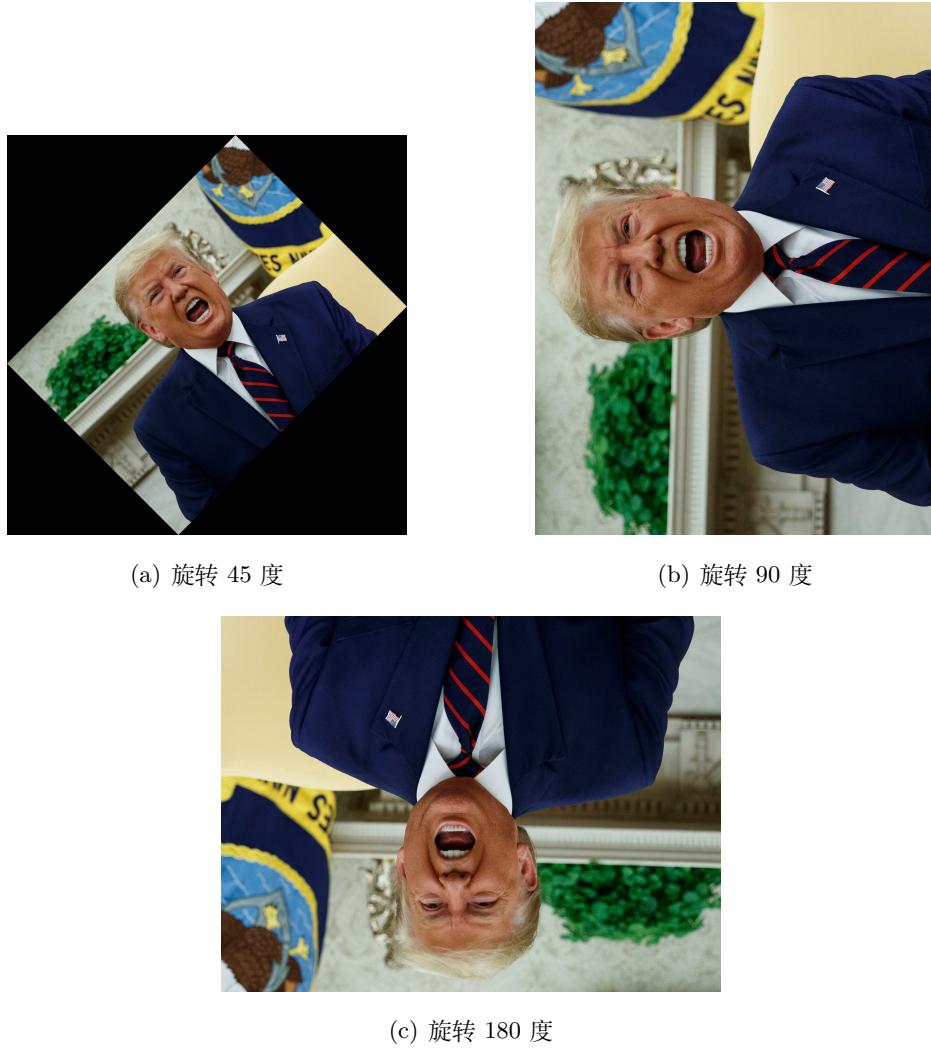


图 7: 旋转

7 主要核心代码

7.1 读取与显示图像

```
# -*- coding:utf-8 -*-
from PIL import Image
im = Image.open("./Trump.jpg")
im.show()
```

7.2 改变图像形式

JPG 图片 RGB 模式转灰度模式

```
# -*- coding:utf-8 -*-
from PIL import Image
im = Image.open("./Trump.jpg");
#转成灰度图像
im1 = im.convert('L')
im1.save('Trump_grey.jpg')
```

JPG 转二值图像

```
# -*- coding:utf-8 -*-
from PIL import Image
im = Image.open("./Trump.jpg")
#转成灰度图像
tmp = im.convert('L')

#设置阈值，大于 threshold 为黑色
threshold = 110
pic = []
#灰度图像每个像素由 0-255 表示
for i in range(256):
    if i < threshold :
        pic.append(0)
    else :
        pic.append(1)
res = tmp.point(pic,'1')
res.save('Trump_binary_110.jpg')
```

7.3 改变图片大小与旋转图片

调整大小

```
# -*- coding:utf-8 -*-
from PIL import Image

outputSize = [
    (320, 240),
    (160, 120),
    (640, 480)
]
im = Image.open("./Trump.jpg");
#改变图片大小
for one in outputSize:
    im.thumbnail(one)
    print(im.format,im.size,im.mode)
    im.save('Trump' + str(one[0]) + '_' + str(one[1]) + '.png')
```

旋转图像

```
# -*- coding:utf-8 -*-
from PIL import Image
import math
im = Image.open("./Trump.jpg")
# 旋转图片

W = im.size[0]
H = im.size[1]
angle = 45

# 计算旋转后的图片大小
newW = math.cos(math.radians(angle)) * W + math.sin(math.radians(angle)) * H
newH = math.cos(math.radians(angle)) * H + math.sin(math.radians(angle)) * W
out = Image.new("RGB", (int(newW), int(newH)))
box = ( int((newW - W) / 2), int((newH - H)/ 2), int((newW + W) / 2), int((newH + H)/ 2))
out.paste(im, box)
out = out.rotate(angle)
out.save('Trump_45.png')

spin90 = im.transpose(Image.ROTATE_90)
spin90.save('Trump_90.png')
```

```
spin90 = im.transpose(Image.ROTATE_180)
spin90.save('Trump_180.png')
```

8 参考资料

参考文献

[1] Fredrik Lundh,Alex Clark.

Python Imaging Library Reference

<https://pillow.readthedocs.io/en/stable/reference/Image.html#PIL.Image.Image.point>

[2] 大蛇王 (Nick name)

python 图片二值化处理（处理后为纯黑白的图片）

<https://blog.csdn.net/t8116189520/article/details/80271804>