

# python 及计算机视觉的基本操作实验报告

张宇鑫 22090032057

2024 年 9 月 11 日

## 1 实验目的

了解 python 基本语法的使用，同时进行基本的计算机视觉操作。

## 2 实验内容

### 2.1 python 基本语法

Python3 版本查看

```
PS C:\Users\86157> python --version
Python 3.8.10
```

Python3 简单语法实现：

数字求和

```
1 num1=input('Enter first number: ')
2 num2=input('Enter second number: ')
3 sum=int(num1)+int(num2)
4 print('{0} + {1} = {2}'.format(*args: num1,num2,sum))
```

```
C:\Users\86157\PycharmProjects\pythonProject8\venv\Scripts\python.exe C:\Users\86157\PycharmProjects\pythonProject8\python3.py
Enter first number: 6
Enter second number: 5
6 + 5 = 11
Process finished with exit code 0
```

for 循环与算法使用

求最大连续子序列和：

暴力求解：

```
#求任意子序列和
arr = [1,6,-5,2,7,3,-8,-4,-9,10]
sum=0
print('输入求和序列首尾元素的下标: ')
m=int(input())
n=int(input())
for i in range(m,n+1):
    sum+=arr[i]
print('序列和为: ',sum)
```

```
输入求和序列首尾元素的下标:
6
2
序列和为:  -12
```

优化，使用前缀和：

```
#求任意子序列和
arr = [1, 6, -5, 2, 7, 3, -8, -4, -9, 10]
prefix = [0] * len(arr)
prefix[0] = arr[0]
for i in range(1, len(arr)):
    prefix[i] = prefix[i-1] + arr[i]
print('输入求和序列首尾元素的下标: ')
m = int(input())
n = int(input())
print('序列和为: ', prefix[n] - prefix[m-1])
```

## 2.2 python 计算机视觉

灰度图像+缩放

```
1 from PIL import Image
2 from numpy import *
3 from pylab import *
4 import os
5 pil = Image.open('img.png').convert('L') #读取图片并转为灰度图
6 pil.thumbnail((256, 256)) #缩放图片
7 pil.show()
```



复制粘贴操作

```

from PIL import Image
from numpy import *
from pylab import *
import os

pil=Image.open('img.png')
box=(10,10,60,60)
region=pil.crop(box)
region = region.transpose(Image.ROTATE_180)
pil.paste(region,box)
pil.show()

```



旋转缩放操作

```

from PIL import Image
from numpy import *
from pylab import *
import os

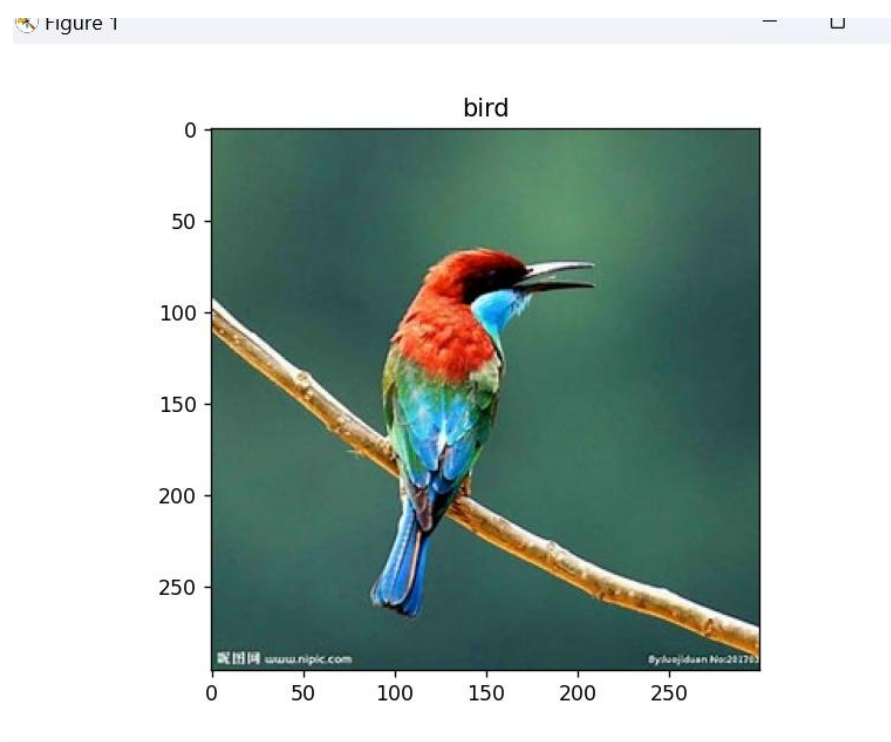
pil=Image.open('img.png')
out = pil.resize((256,256))
out = out.rotate(45)
out.show()

```



绘制坐标轴+标题

```
from PIL import Image
from numpy import *
from pylab import *
import os
arr=array(Image.open('img.png'))
imshow(arr)
title('bird')
show()
```



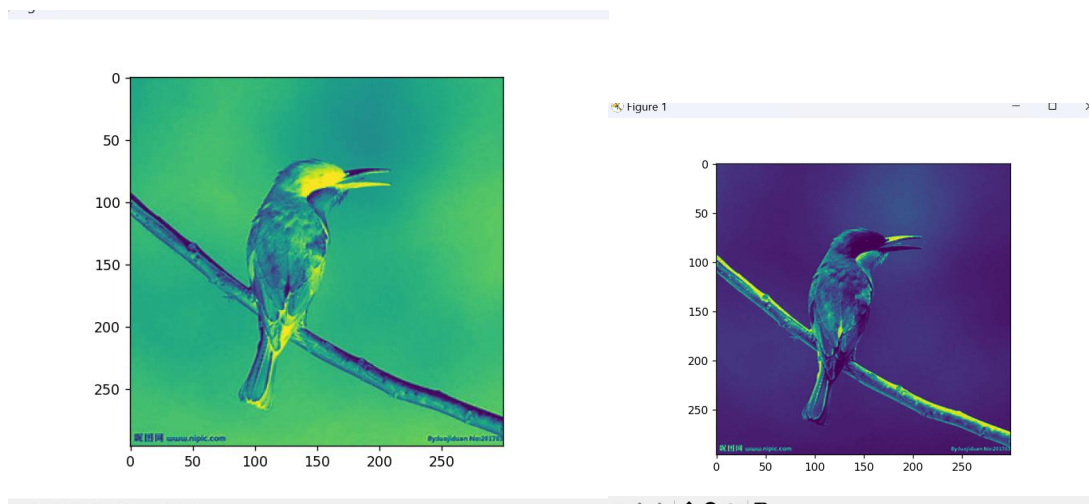
轮廓

```
1 from PIL import Image
2 from numpy import *
3 from pylab import *
4 import os
5 arr=array(Image.open('img.png').convert('L'))
6 figure()
7 gray()
8 contour(*args: arr,origin='image')
9 axis('equal')
10 axis('off')
11 show()
```

Figure 1

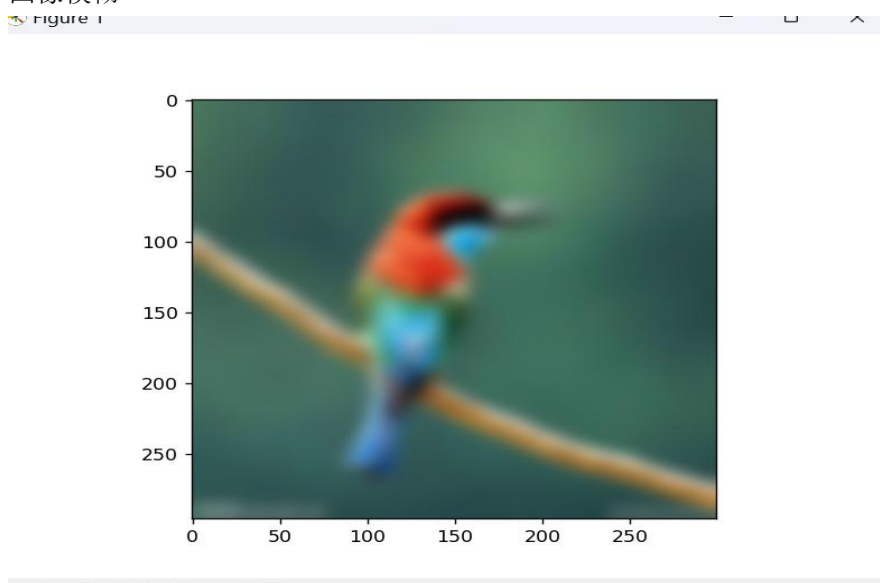


灰度变换



```
1  from PIL import Image
2  from numpy import *
3  from pylab import *
4  import os
5  arr=array(Image.open('img.png').convert('L'))
6  arr2=255-arr
7  arr3=255.0*(arr/255.0)**2
8  imshow(arr2)
9
10 show()
```

图像模糊





```
1  from PIL import Image
2  from numpy import *
3  from pylab import *
4  from scipy.ndimage import filters
5  import os
6  im = array(Image.open('img.png'))
7  im2 = zeros(im.shape)
8  for i in range(3):
9      im2[:, :, i] = filters.gaussian_filter(im[:, :, i], 5)
10 im2 = uint8(im2)
11 imshow(im2)
12 show()
```

### 3 实验结果

利用 python 的基本语法，使用了 for 循环完成了一个简单的求任意序列和的问题，同时利用 fittencode 工具帮助我优化代码，实现了更低复杂度的 python 程序。然后通过使用 python 视觉的基本内容，完成了诸如灰度调节、模糊度调节、图片旋转、放缩等基本操作。

### 4 实验感悟

通过本次实验，我对于 python 基本的语法内容有了更深刻的理解与掌握，同时也了解到了 python 视觉的基本知识，掌握了很多对于图片处理的内容，通过对知识点的掌握，也使我对于 python 视觉有了更加好奇的感觉，通过代码的编写实现图片的改变，使得我对于计算机视觉有了更加深入的了解。