

计算机视觉第一次作业

1611640305__温吉祥

一:实验内容:

理解图像基本处理原理并使用 Python 实现，主要实现以下三种操作：

- 1: 逆时针 θ 度进行图片旋转
- 2: 指定方向和大小进行图片缩放
- 3: 指定方向和大小进行图片平移

二：工具选择：Python 中图像处理库主要有 PIL、opencv、matplotlib 和 skimage 等，在这里我选择了使用 PIL 和 opencv 两种方法实现，最后再用 matplotlib 以子图形式输出。

三：代码实现与结果展示

1: 查看原图信息



```
import cv2
from PIL import Image
pil_read = Image.open('./gulfstream.png')    #使用 PIL 读取图片
cv_read = cv2.imread('./gulfstream.png')     #使用 cv2 读取图片
pil_read.info --> {'srgb': 0, 'dpi': (72, 72)}
cv_read.shape --> (423, 794, 3)
```

可知原图是一张宽 794, 高 423 的三通道 SRGB 彩图

2: 逆时针 θ 度进行图片旋转

#rotate 函数

```
def rotate(im1,im2,angle,width,height):  
    """  
    im1:使用 PIL 读取的图片  
    im2:使用 cv2 读取的图片  
    angle:旋转角度  
    width:图片宽度  
    height:图片高度  
    """  
  
    #使用 PIL 库的 rotate 函数旋转  
    pil_rotated = im1.rotate(angle)  
    #先通过 getRotationMatrix2D 函数得到图像的旋转矩阵  
    temp = cv2.getRotationMatrix2D((width//2,height//2),angle,1)  
    #再通过仿射变换函数 warpAffine 得到旋转后的图像  
    cv_rotate = cv2.warpAffine(im2,temp,(width,height))  
    #将 opencv 中的 BGR 格式转换成 RGB 格式,方便后续使用 matplotlib 正常显示图  
    片  
    cv_rotated = cv2.cvtColor(cv_rotate,cv2.COLOR_BGR2RGB)  
    #返回两种方法得到的结果  
    return pil_rotated,cv_rotated
```

结果展示:

下图为旋转 45° 所得结果



3: 指定方向和大小进行图片缩放

```
#scale
def scale(im1,im2,scale_ratio,width,height):
    """
    scale_ratio:缩放比例列表
    Image.ANTIALIAS:抗锯齿处理
    """
    pil_scaled = im1.resize((width//scale_ratio[0],
                             height//scale_ratio[1]),Image.ANTIALIAS)
    cv_scale = cv2.resize(im2,(width//scale_ratio[0],
                             height//scale_ratio[1]))
    cv_scaled = cv2.cvtColor(cv_scale,cv2.COLOR_BGR2RGB)
    return pil_scaled,cv_scaled
```

结果展示:

下图为沿 X 轴缩放 1/3,Y 轴缩放 1/5 得到的结果,
明显发现左图做了边缘抗锯齿处理

PIL沿X轴缩放1/3, Y轴缩放1/5



cv2沿X轴缩放1/3, Y轴缩放1/5



4: 指定方向和大小进行图片平移

```
def shift(im1,im2,shift_size,width,height):
    """
    M:自定义平移矩阵
    """
    pil_shifted = ImageChops.offset(im1,width//shift_size[0],
                                     height//shift_size[1])
    M = np.float32([[1,0,width//shift_size[0]],
                    [0,1,height//shift_size[1]]])
    cv_shift = cv2.warpAffine(im2,M,(width,height))
    cv_shifted = cv2.cvtColor(cv_shift,cv2.COLOR_BGR2RGB)
    return pil_shifted,cv_shifted
```

结果展示：

下图为沿 X 轴平移 $1/8$, Y 轴平移 $1/6$ 得到的结果

PIL沿X轴平移 $1/8$, Y轴平移 $1/6$



cv2沿X轴平移 $1/8$, Y轴平移 $1/6$



以下是所有图片放一起的效果

原图



PIL逆时针旋转 45°



cv2逆时针旋转 45°



PIL沿X轴缩放 $1/3$, Y轴缩放 $1/5$



cv2沿X轴缩放 $1/3$, Y轴缩放 $1/5$



PIL沿X轴平移 $1/8$, Y轴平移 $1/6$



cv2沿X轴平移 $1/8$, Y轴平移 $1/6$



