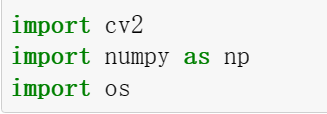
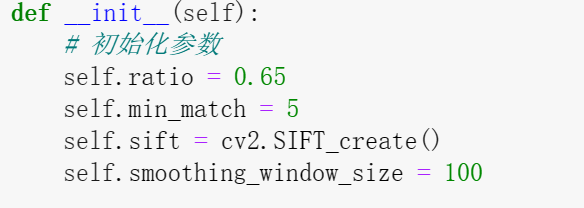
# 图像拼接-实验报告

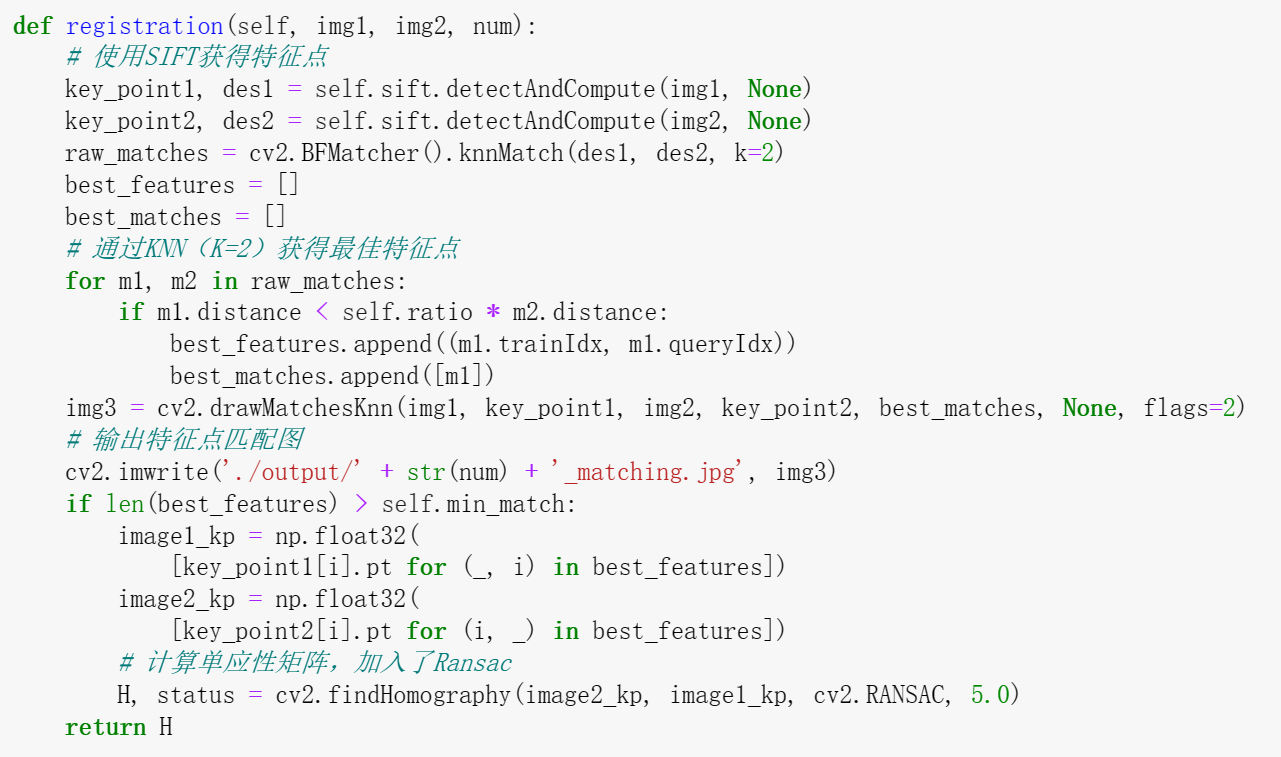
1. 实现说明



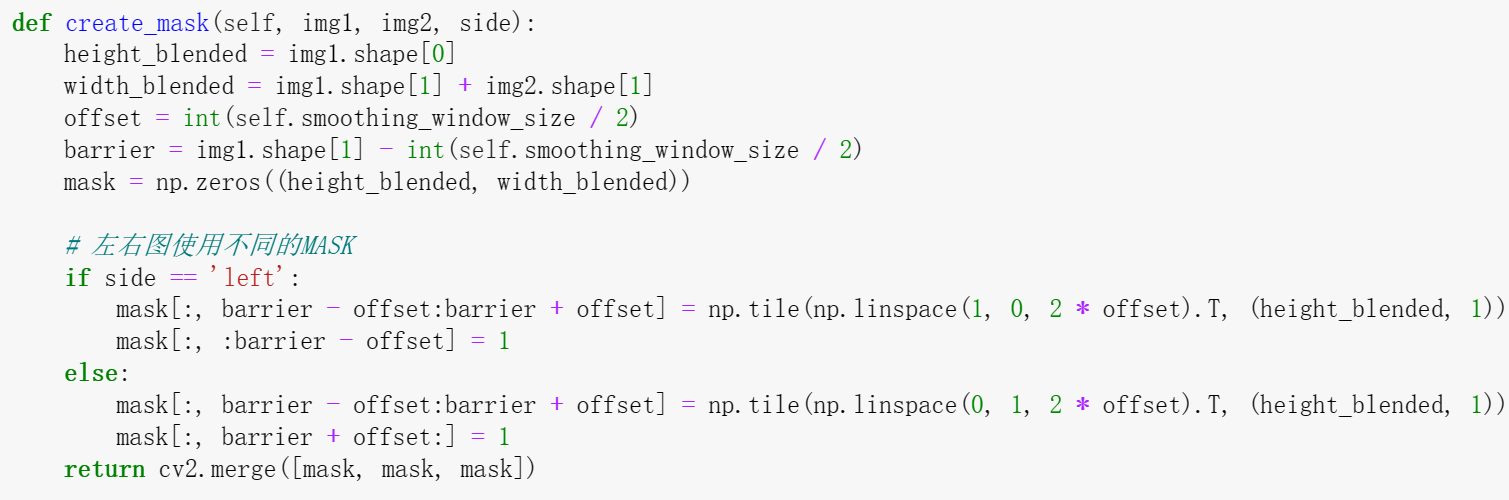
首先导入相关的库



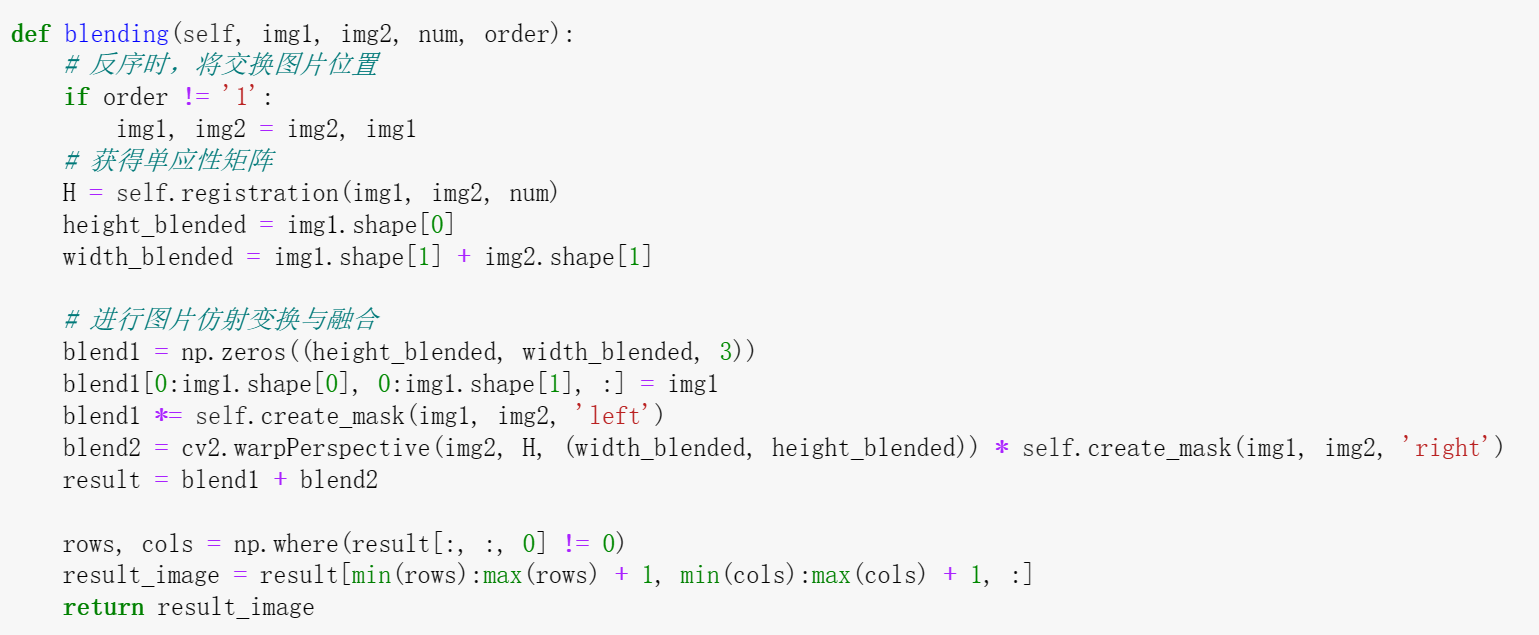
首先定义了一个类，用于图像拼接。在初始化函数中，设置了一些参数，如特征点匹配的阈值、最小匹配点数、SIFT算法对象和平滑窗口大小



这是图像注册函数。首先，使用SIFT算法检测图像的特征点，并计算特征描述子。然后，使用暴力匹配器（cv2.BFMatcher）进行特征点匹配，并根据设定的阈值（ratio）筛选出最佳的匹配点。接着，利用匹配点绘制特征点匹配图，并保存在输出文件夹中。最后，根据最佳匹配点计算单应性矩阵，使用RANSAC算法进行鲁棒估计，返回单应性矩阵H。



这个函数用于创建遮罩，用于在图像融合时控制边缘效果。根据传入的参数side，分别生成左边和右边的遮罩。遮罩是一个矩阵，对应图像的每个像素点，控制在融合过程中该像素点的权重。



这个函数用于图像融合。首先根据参数order判断是否需要交换图像的顺序，然后调用registration函数获取单应性矩阵H。接着，创建一个空白的融合图像blend1，将第一张图像插入到左边，并根据左边的遮罩控制融合。

1. 结果截图



这是用于测试的三张图片，在鼎新楼下拍的照片

结果1：两张图片拼接：



结果2：三张图片拼接



1. 运行说明

实验环境为在anaconda中创建的虚拟环境，python版本为3.7，以及对应版本的opencv、numpy等包.

准备一个包含若干图片的文件夹即可，运行以下代码：结果存放在output文件夹中.

