**图像拼接中的单应性变换**

**一、实验目的**

单应性变换是一种重要的几何变换方法，用于描述两个平面之间的点对应关系。在本实验中，将通过特征点检测、特征点匹配、单应性矩阵计算以及图像变换，完成两张图像的拼接。

**二、实验原理**

单应性变换是一种用于将一个平面上的点映射到另一个平面上的点的几何变换。它由一个3\*3的矩阵H表示，可以将图像中的一个点（x1，y1）通过以下方式映射到另一个点（x2，y2）：

其中，单应性矩阵H表示为：

单应性矩阵H 的求解需要至少四对对应的特征点。在实际应用中，通常通过特征点检测算法（如SIFT）提取图像中的特征点，并通过特征点匹配算法（如FLANN）找到两幅图像中的对应点对，最后利用RANSAC算法计算单应性矩阵。单应性变换在图像处理中有广泛的应用，例如：

图像拼接：通过单应性变换将多张图像拼接成一张全景图。

视角变换：将图像从一个视角变换到另一个视角。

图像校正：校正图像中的透视畸变，使其看起来像从正面拍摄的图像。

增强现实：在真实世界图像中叠加虚拟对象，并使虚拟对象在正确的透视关系中显示。

**三、实验步骤**

1.导入必要的库

2.读取输入图片。读取两张待拼接的图像 imageA 和 imageB。使用 cv2.imread 函数将图像文件加载到内存中。

3.创建SIFT对象并提取特征点与描述子。SIFT是一种常用的特征点检测和描述算法，能够提取图像中的关键特征点及其周围的描述子，用于后续的匹配和变换。

4.创建FLANN匹配器并执行knn匹配。FLANN是一种高效的近似最近邻搜索算法，适合于大数据集中的快速搜索。FLANN匹配器的创建需要两个主要参数：indexParams，用于指定索引树的构建方式。在这里选择了KD树，一种用于最近邻搜索的树形数据结构。searchParams，用于指定搜索的参数，包括每次搜索中检查的树的数量。knn匹配是一种寻找最近邻居的方法。在这里，选择 k=2，表示为每个特征点找到两个最近的邻居。

5.过滤特征点。在进行knn匹配后，需要过滤匹配结果以保留最好的匹配对。通过比较距离，保留距离比阈值（0.75）小的匹配点对。这一步旨在排除错误匹配，保留更可靠的匹配点对。

6.计算视角变换矩阵与图像拼接。如果筛选后的匹配点对超过10对，使用RANSAC来计算单应性矩阵。RANSAC算法可以有效地排除误匹配，计算出更加准确的单应性矩阵。调用cv2.findHomography函数计算单应性矩阵。使用这个矩阵，可以将图像B进行透视变换，使其与图像A对齐。调用cv2.warpPerspective函数实现这个变换。透视变换会根据单应性矩阵重新计算图像B中的每个像素的位置，使其在图像A的坐标系中呈现出正确的透视关系。

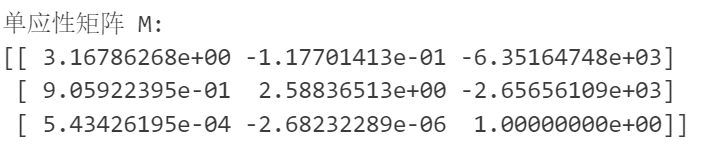
7.计算重叠区域。为了进行图像的无缝拼接，需要计算两幅图像的重叠区域。在这里，通过从左到右和从右到左的扫描，找出重叠区域的左右边界。

8.加权处理重叠区域。在重叠区域，采用加权混合的方法进行图像融合，使拼接区域过渡更加自然。对于每个像素，计算其到重叠区域左右边界的距离，根据距离比例计算加权值，进行像素值的加权平均。

9.保存结果。

**四、实验结果**

**单应性矩阵**

****

**拼接结果**

原图左 原图右



拼接图

**五、结果分析**

在特征点匹配中，使用SIFT和FLANN匹配器，在两幅图像中找到了足够的匹配点对，确保了单应性矩阵的计算精度。在重叠区域处理时，通过加权混合处理，平滑了图像重叠区域，减小了拼接线的明显性。拼接后的图像整体效果较好，但在某些复杂场景下，可能还需要进一步优化特征点匹配和变换过程。