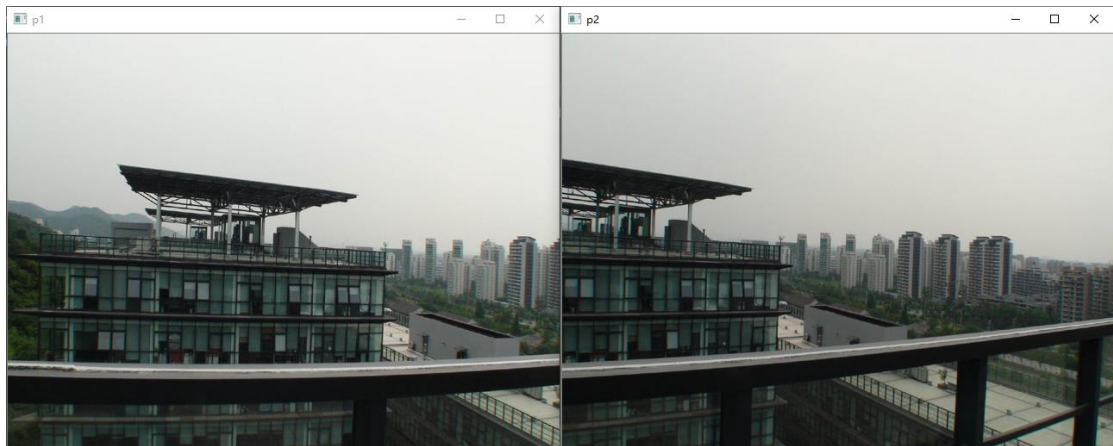


图像拼接算法预研

1. 概论

图像拼接技术就是将数张有重叠部分的图像拼成一幅大型的无缝高分辨率图像的技术。将多张拍摄自不同时间、不同视角甚至是不同的传感器的影像根据其重叠部分拼接起来。



图像拼接所要研究的问题主要有两个——图像配准与图像融合。图像配准目前比较流行的方法可以分为两大类——基于特征点与基于区域的方法。

基于区域的方法其实现算法较为简单，然而局限性很大。它的计算量很大，另外该方法对尺度、亮度以及旋转变化的敏感性，一般只适用于平移。而基于特征点的配准方法其速度快，并且对于灰度变化有较强鲁棒性，不过，精确度和稳定性还是要寻找更佳的方法来解决。基于特征点的方法的关键在于寻找特征点，较为常见的算法有 ORB 特征点检测法与 SURF 特征点检测法，这两种算法将在下文详细介绍。

图像融合目前的方法都比较简单，普遍使用线性加权平均法。由于图像的光照随机性比较大，要找出一种统一的调整光照的方法相当困难。

2. 图像配准算法类别

2.1 基于区域的方法

基于区域的方法又称为基于模板匹配的图像拼接。基于模板的图像拼接特点和适用范围：图像有重合区域，且待拼接图像之间无明显尺度变化和畸变。常用实例：两个相邻相机水平拍摄图像拼接。其实际过程如下：

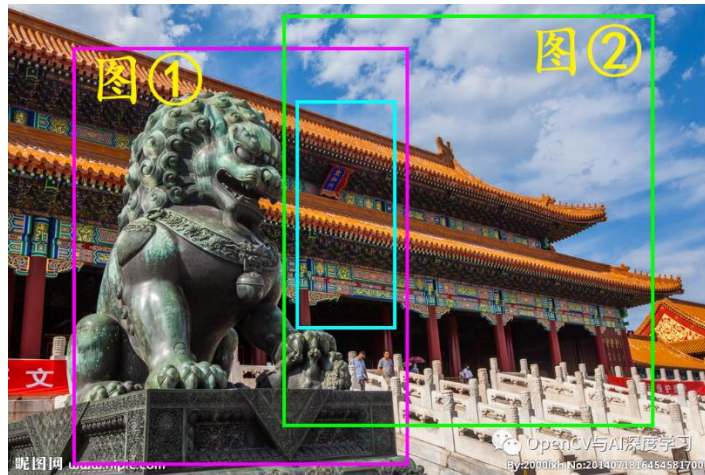


图 1. 基于区域的方法 demo

假设在上面这张图像中，我们通过拍摄获得图 1 和图 2 的内容，其中青蓝色的区域是两张图像的 ROI 区域，通过 ROI 区域对两张图像进行重定位，最终就能得到融合后的全景图。

但是不难发现，基于区域的方法具有很强的局限性，首先是拼接图像之间的重合区域不能有明显的畸变，但是对于一些广角相机来说，在边缘的拼接处存在的畸变是无法忽视的；其次，用来计算两张图像之间重叠区域的过程具有很高的复杂度，因此在算法实现上所需要的耗时也是该算法的一个缺点。

2.2 基于特征点的方法

基于特征点的方法是通过两幅图像进行计算，获取各自的特征点，之后通过特征点描述后进行匹配，将左右两图的特征点坐标关系进行计算，最终通过投射关系得到两图的空间位置关系。常用的特征点有 SIFT, ORB, SURF, Harris 角点等。考虑到计算所需耗时和效率，通常会选择使用 SURF 和 ORB 等高效的特征点进行匹配计算。如下图：

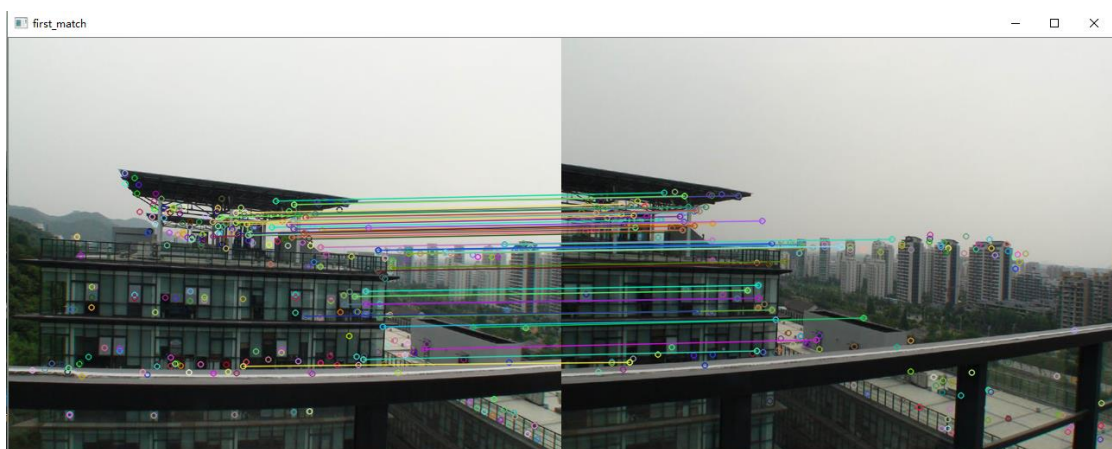


图 2. SURF 匹配结果图

其匹配的过程包括：

- (1) 提取特征点

提取特征点可以理解为在单张图像中获取最能代表这张图像的信息

(2) 特征点描述

特征点描述是指通过一种数值表示将图像中的特征点转换为可供比较的向量或描述符。这些描述符捕捉了特征点周围的局部信息，使得相似的特征点在特征空间中距离较近，便于进行特征点的匹配、识别和跟踪。获取优秀匹配点。

(3) 获取优秀匹配点

在图像匹配中，获取优秀匹配点意味着从所有的匹配点中选择那些质量较高、相似度较高的匹配点，以排除误匹配。这有助于确保匹配结果更准确，从而在诸如目标跟踪、图像配准等应用中获得更好的效果。

(4) 图像配准

图像配准的目的是为了将两张图片映射到同一坐标系，方便后面的拼接处理。

以上所有功能在 OpenCV 中都已存在对应函数。

3. 图像融合算法

图像融合算法部分只介绍简单的图像拷贝 + 去裂缝处理方法，因为这是目前图像拼接过程当中比较常见的问题。当我们获得两张同意坐标系下的图像后，首先会进行图像拷贝，即将左图直接拷贝到配准图上。但是这样可能会产生如下的效果：

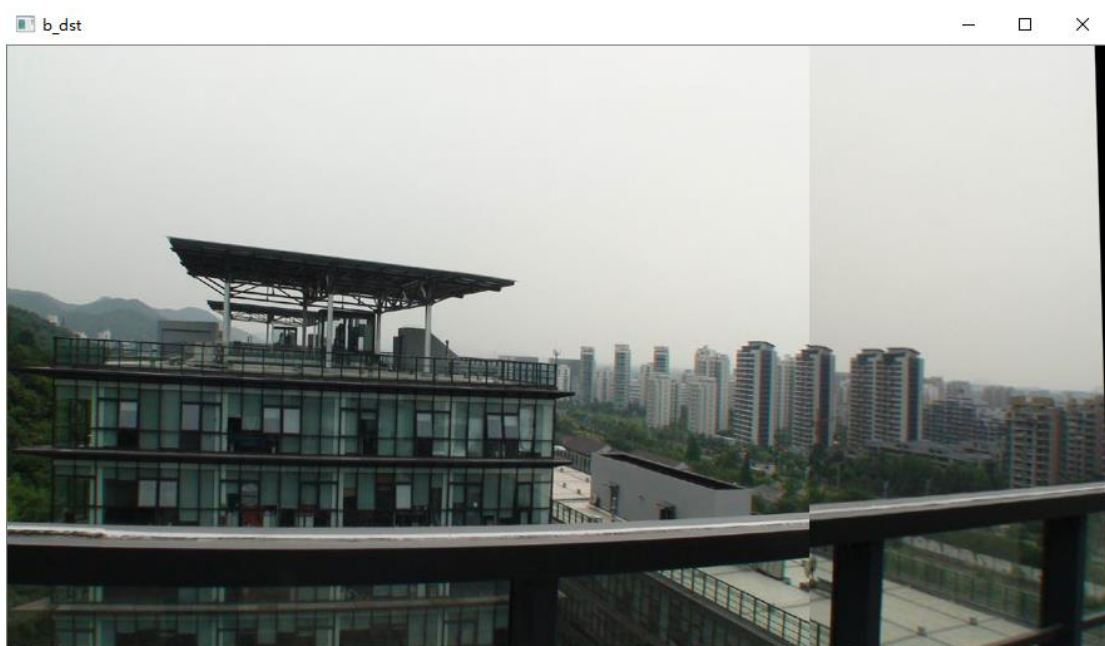


图 3. 未处理裂纹的融合图像

在拼接部分，会显得十分生硬，并且在扶手处还存在拼接无法衔接的问题。

因为在拼接图的交界处，两图因为光照色泽的原因使得两图交界处的过渡很糟糕，所以需要特定的处理解决这种不自然。这里的处理思路是加权融合，在重叠部分由前一幅图像慢慢过渡到第二幅图像，即将图像的重叠区域的像素值按一定的权值相加合成新的图像。优化后的图像如下：

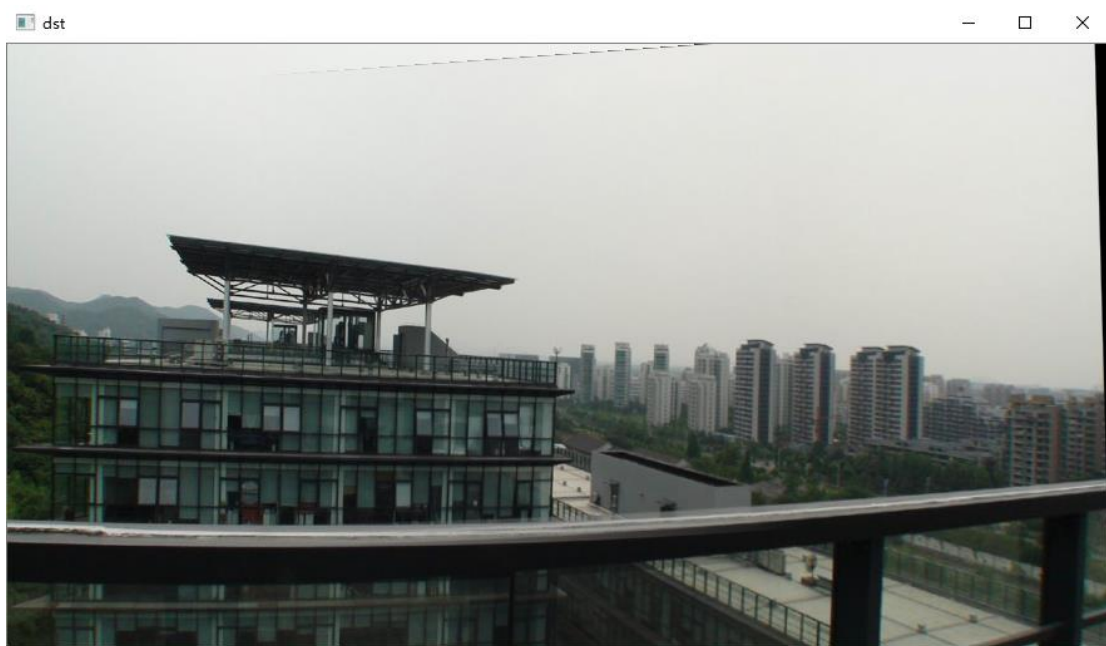


图 4. 处理裂纹后的融合图像