# **实验三：全景图**

组员：梁耀宗 李阔

# **1.实验环境**

Python3.8

# **2.实验内容**

1) 将图像变形到球面坐标

2) 提取特征并进行特征匹配

3) 使用RANSAC对齐相邻图像对

4) 写出相邻图像对之间的变换矩阵

5) 纠正漂移

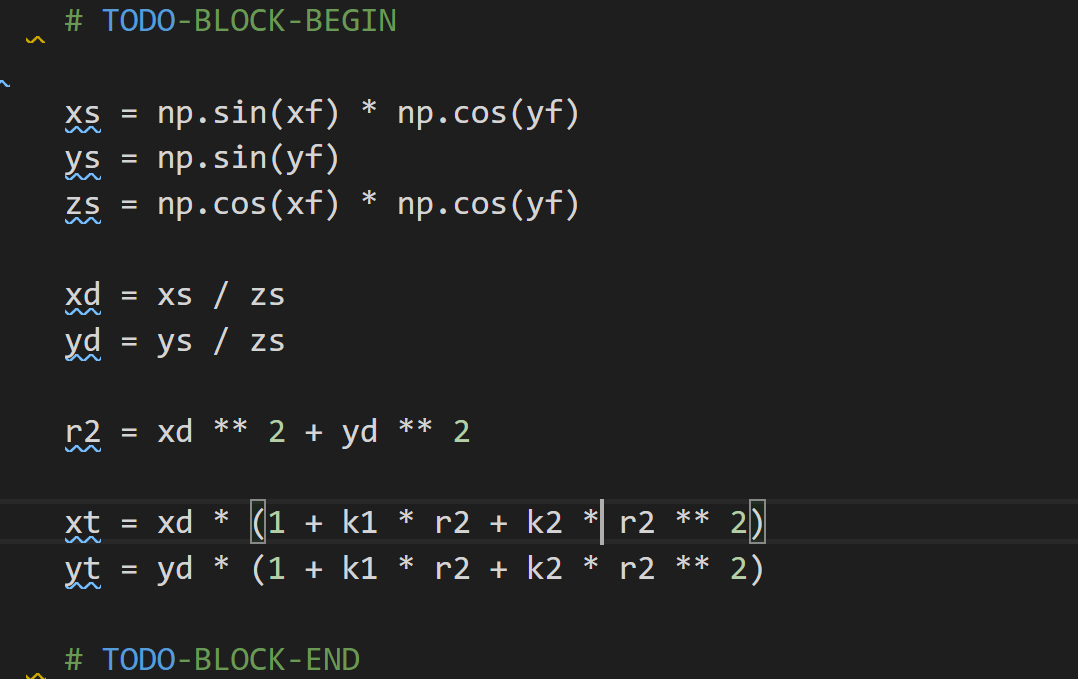
6) 将所有图像融合

7) 裁剪结果

# **3.实现细节**

1) 将每张图像变形为球面坐标

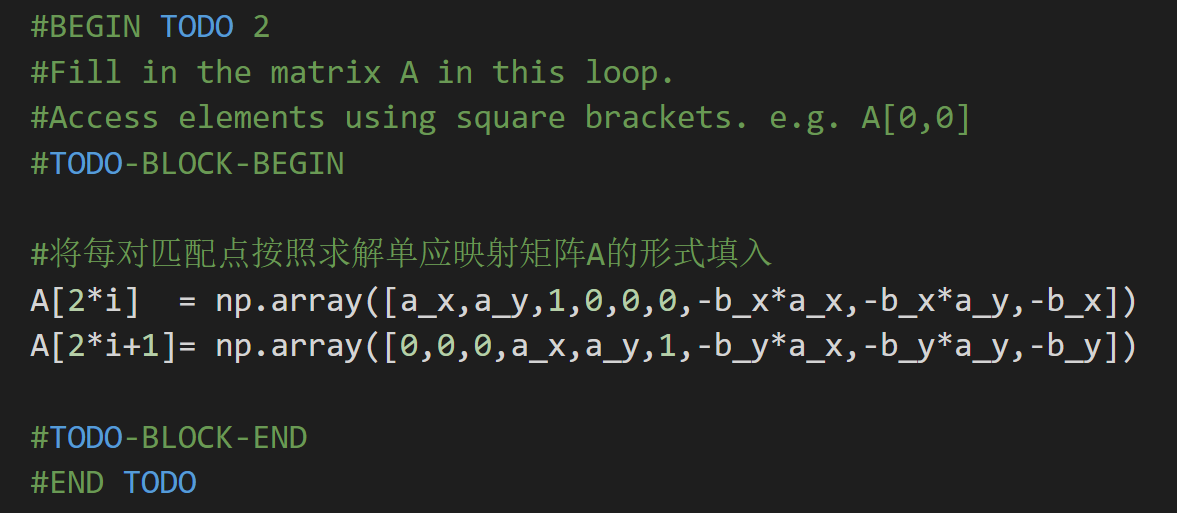
按照球面坐标变换公式，将图像映射到球面上，并根据矫正畸变公式矫正径向畸变

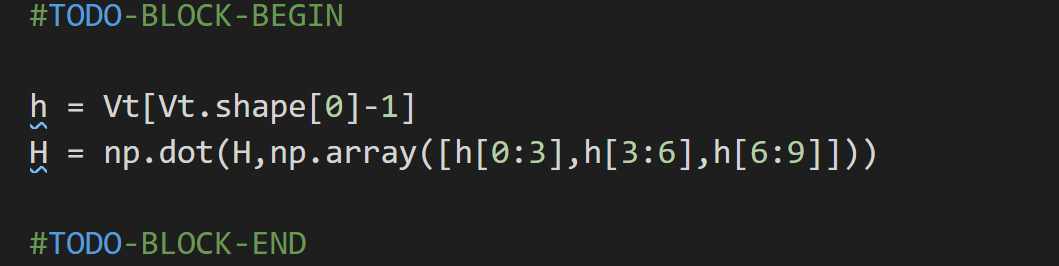


2) 对齐相邻图像对

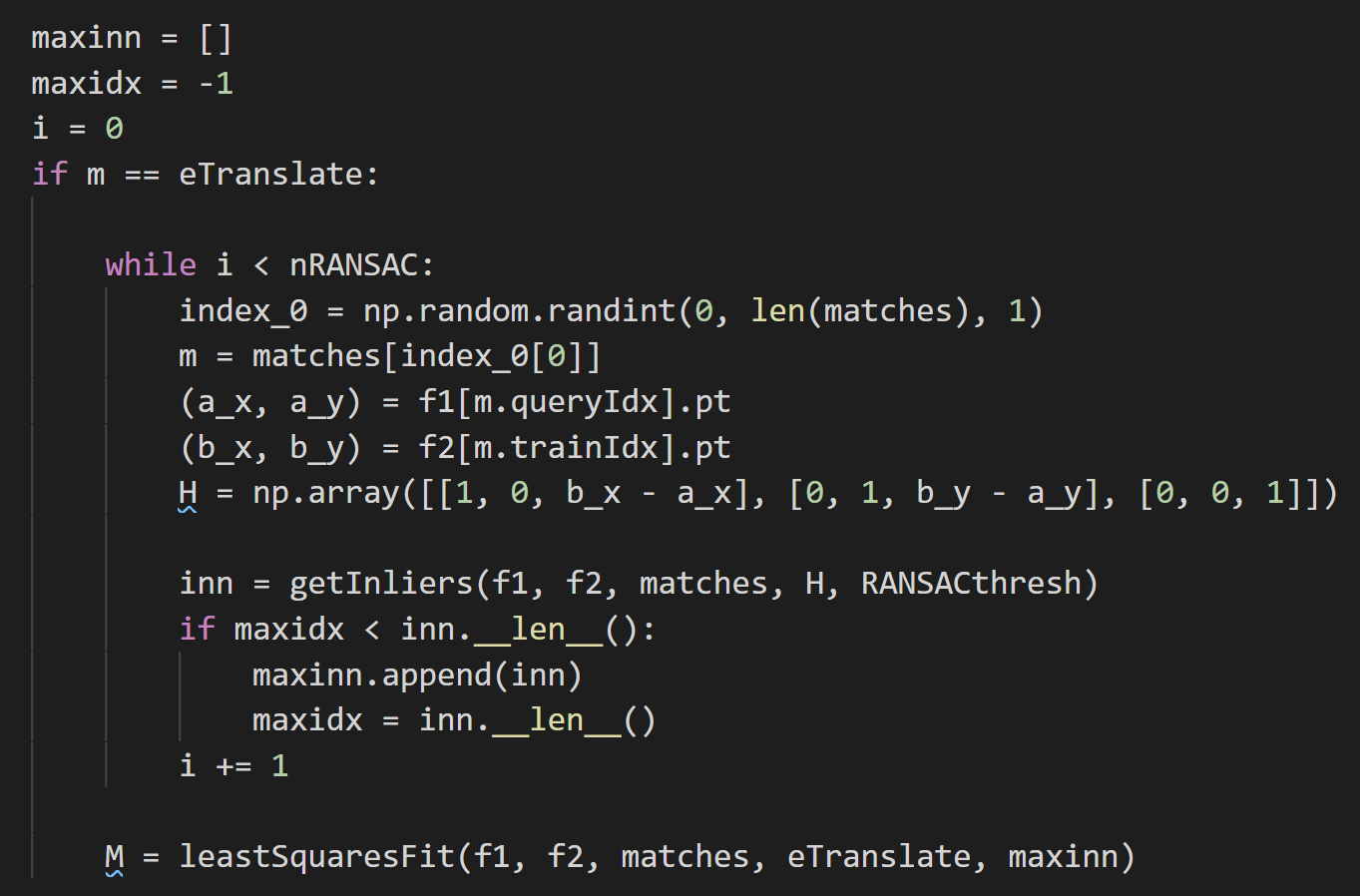
通过SVD奇异值分解计算最佳拟合的单应映射

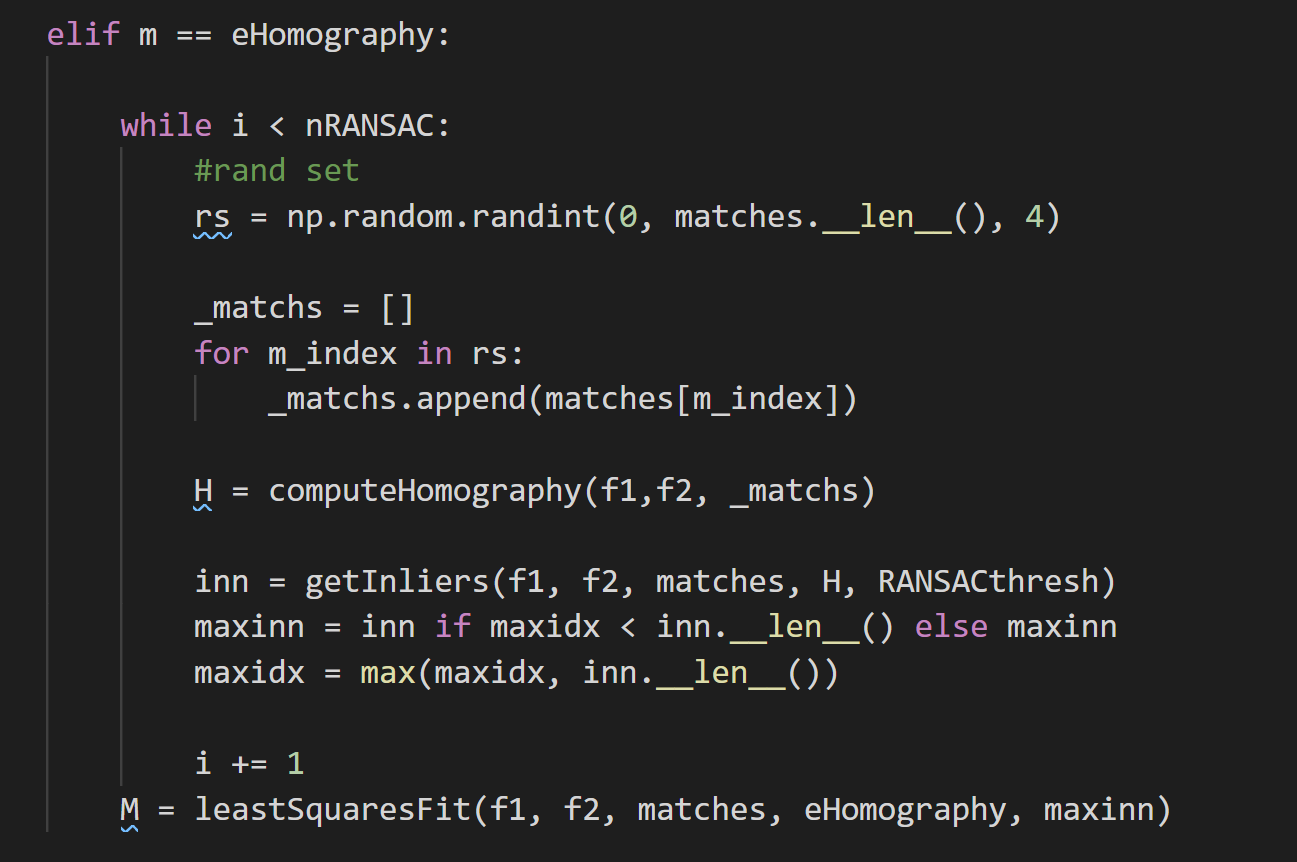
首先按照求解单应映射矩阵形式填充A矩阵，SVD分解用将合适的元素填充至单应映射矩阵H中解得H。





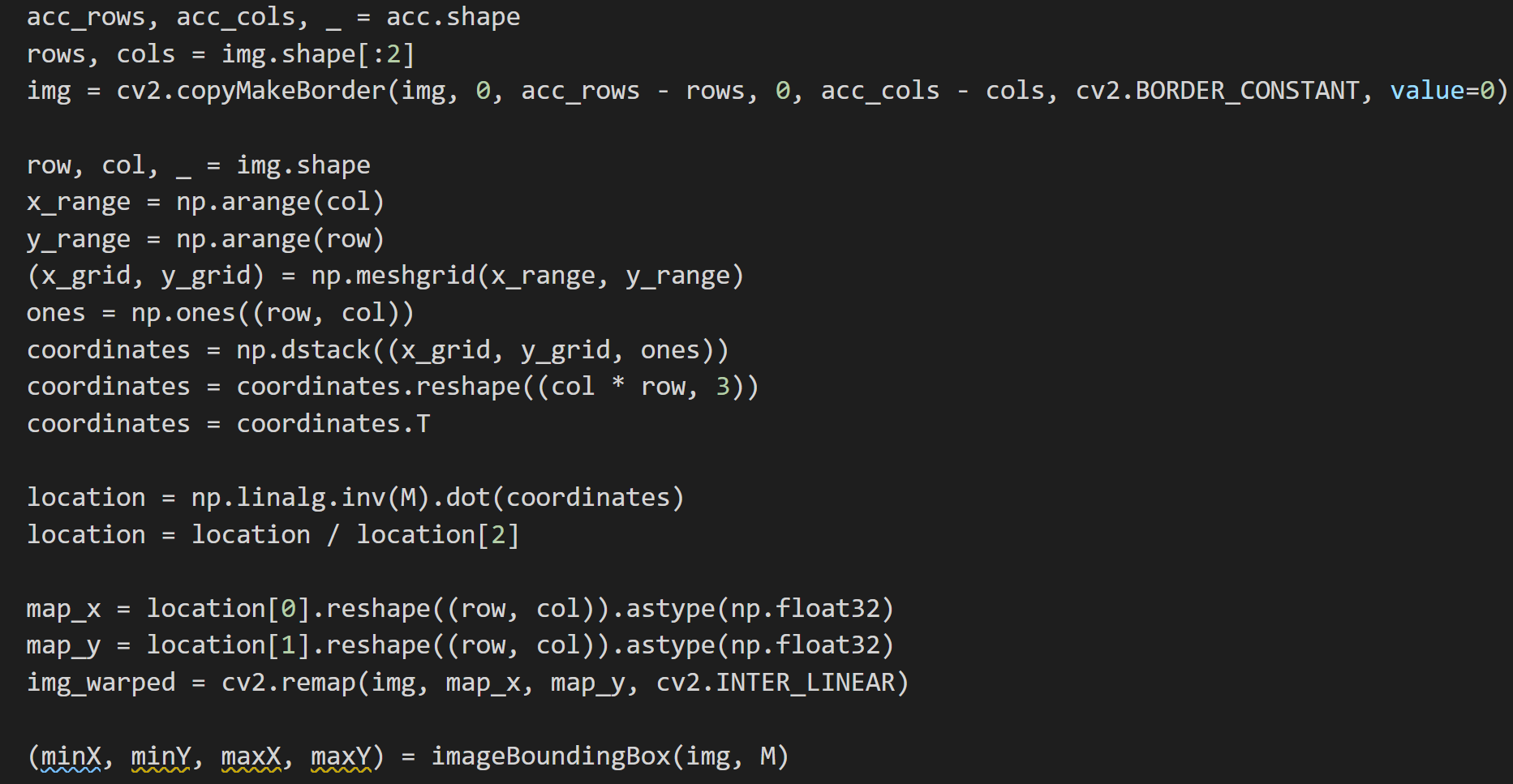
之后根据RANSAC估计两幅图之间的最佳匹配，对于平移变换和单应映射分情况求解。其中，getInliers()获得欧式距离在阈值范围内的匹配点的个数，RANSAC选择匹配点数最多的匹配，leastSquareFit()根据上诉匹配点计算出图像1到图像2的变换矩阵。



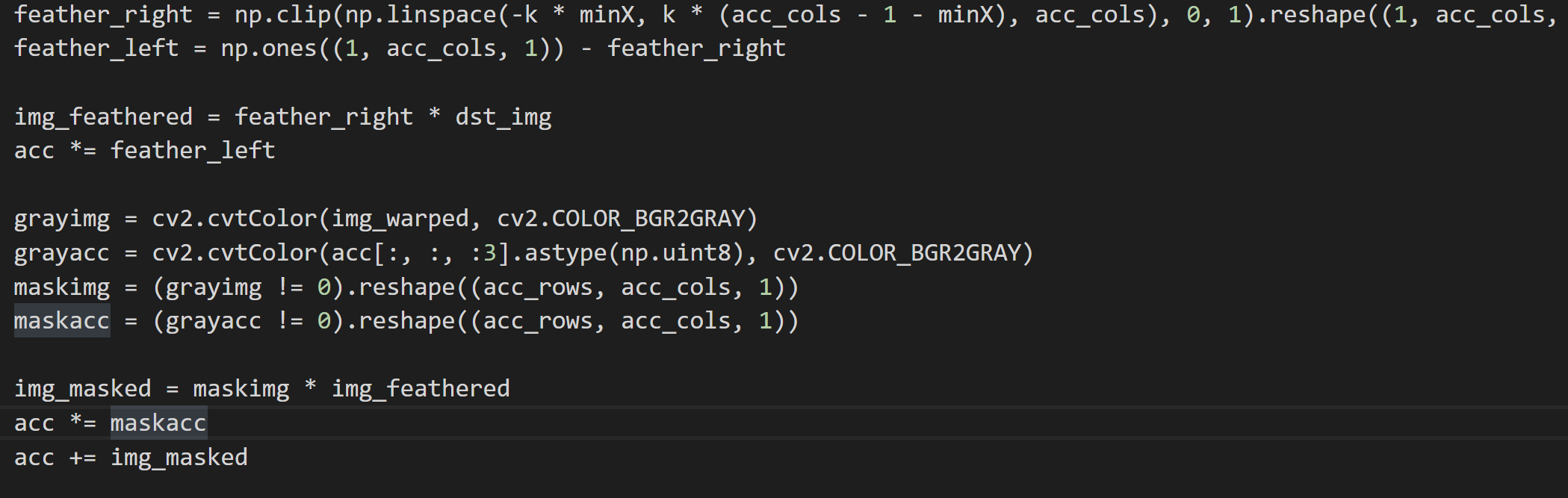


3) 融合并裁剪生成的对齐图像

根据给定的图像和变换矩阵，求出应用变换后的图像的框，根据边框计算出最终拼接图的大小及他们在全景图中的绝对位移，然后将每个图象重新映射到其最终位置，并将其与相邻图像融合，其中，使用羽化功能作为加权函数。

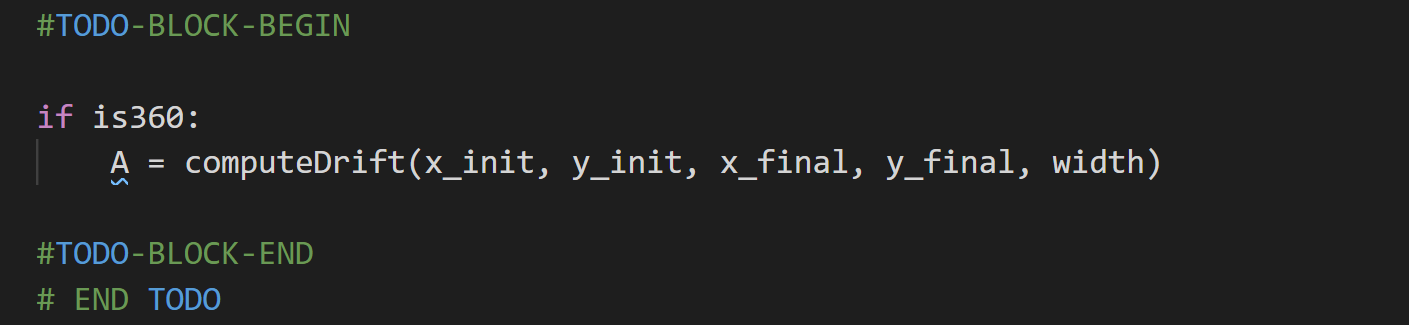


使用简单的羽化功能作为加权函数



4) 消除漂移

如果是360度全景照片，要使左右边缘具有完美的接缝效果。第一幅图像将出现在凭借序列的最左端和最右端。对图像使用线性变换以消除第一个和最后一个图像之间的任何“垂直漂移”



# **4.实验结果**

**应用单应映射对yosemite数据集生成的全景图**

****

**应用平移变换对yosemite数据集生成的全景图**

**山上的雪山

描述已自动生成**

**应用平移变换对campus数据集生成的360度全景图**

**草地上的建筑

描述已自动生成**

# **5.总结与收获**

**通过本次实验，加深了对numpy等常用库的理解与掌握，深刻理解了全景图生成与制作的整个过程，对于过程中使用的矩阵运算和RANSAC等算法有了详细的了解，并且加强了自主学习能力与阅读英文文献的能力，此次实验让我受益匪浅。**

**工作量比例：梁耀宗 18030100139 50%**

**李阔 18030100224 50%**