**实验三：全景图**

**本实验单人完成**

# 实验环境

python：3.7.4

numpy：1.16.4

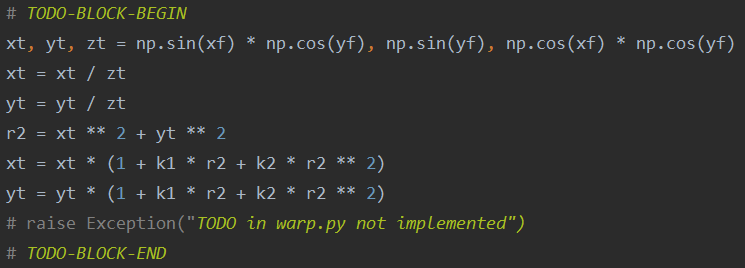
opencv：3.4.2

pillow：6.1.0

pytest：5.0.1

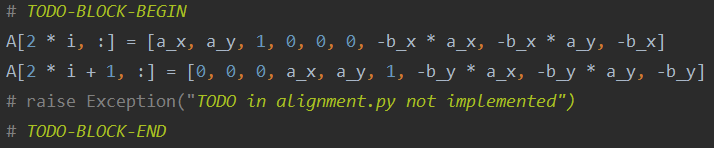
# 实现过程

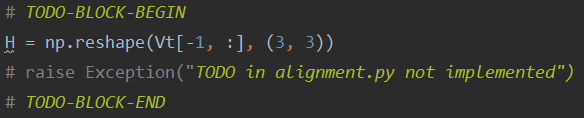
## [TODO1]实现computeSphericalWarpMappings函数



* 首先将极坐标转换为球面坐标，并进行归一化处理（转换为单位球面坐标）。
* 再对归一化后的球面坐标应用径向畸变模型修正径向畸变失真。

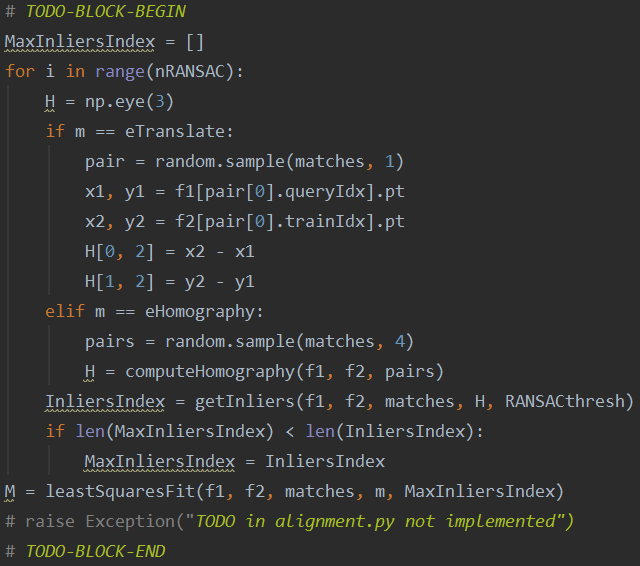
## [TODO2&TODO3] 实现computeHomography函数





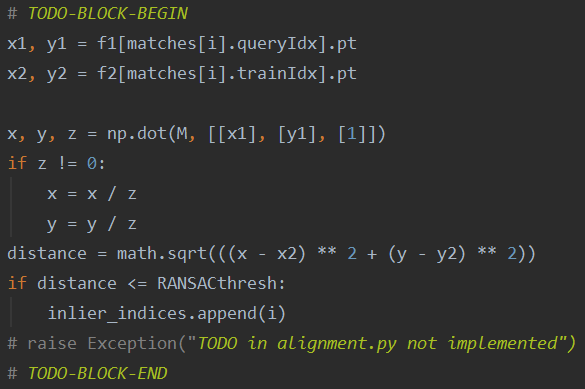
* 首先根据给定的匹配对的坐标通过切片操作给Ah=0的A矩阵进行初始化。
* 对A^TA进行奇异值分解，取右奇异向量的最后一行并调用np.reshape函数将其转换为3\*3矩阵，该矩阵即为单应映射矩阵h的解。

## [TODO4]实现alignPair函数



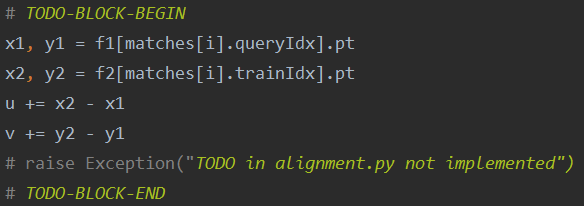
* 首先通过一组匹配对（平移变换每组一个匹配对，单应映射每组四个匹配对）估计变换矩阵H。
* 因为在使用最小二乘法计算变换矩阵M时，离群点对于M的影响很大，所以使用RANSAC方法找出在阈值内的内点再来计算变换矩阵M。调用InliersIndex函数和估计的H矩阵计算出在阈值内的内点在匹配对中的索引值。
* 调用leastSquaresFit函数，利用在阈值内的内点计算出最佳变换矩阵M。

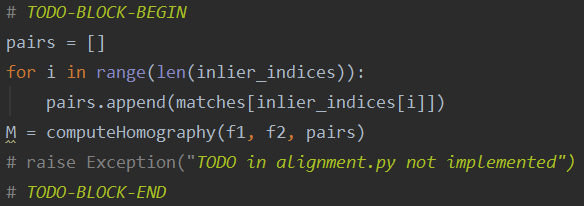
## [TODO5]实现getInliers函数



* 遍历所有匹配对，计算匹配对中两个点变换后的坐标。
* 对坐标进行归一化后计算两个点的欧氏距离。
* 筛选出欧式距离小于阈值的匹配对的索引。

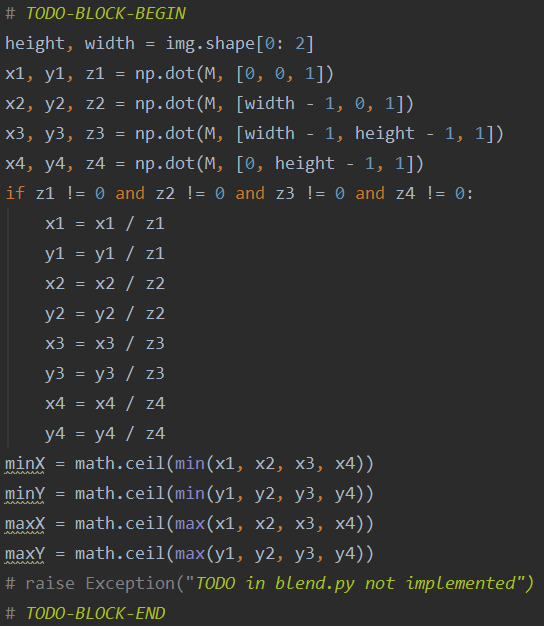
## [TODO6&TODO7]实现leastSquaresFit函数





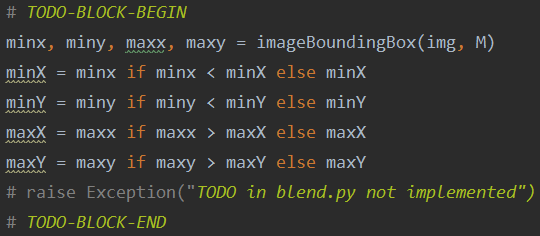
* leastSquaresFit函数通过内点来计算最佳变换矩阵M。
* 对于平移变换，仅需简单计算X和Y坐标的平均平移量。
* 对于单应映射，调用TODO2&TODO3已经实现的computeHomography函数计算M。

## [TODO8]实现imageBoundingBox函数



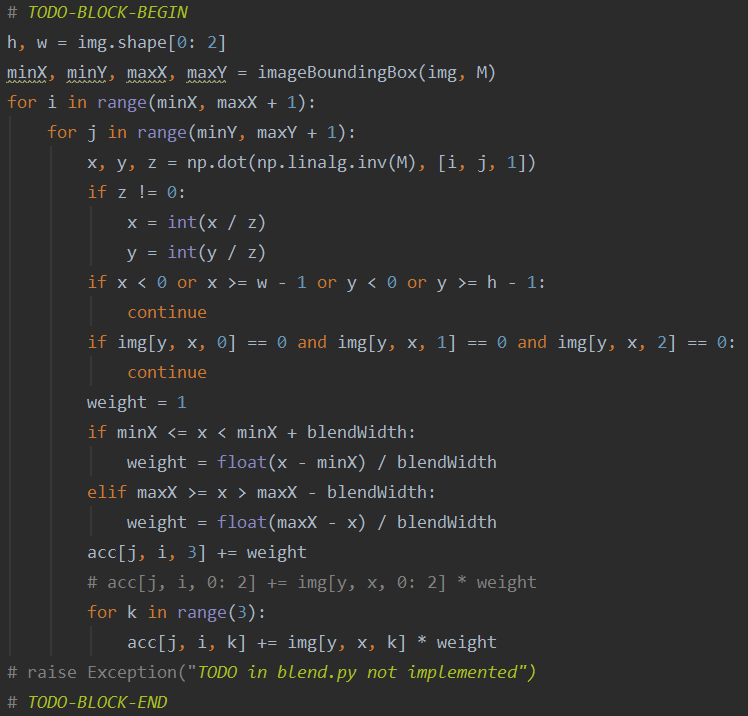
* 由于单应映射变换保持图形的直线性，所以仅需要对于图像的四个顶点应用最佳变换矩阵M。
* 对变换后的坐标进行归一化后进行大小比较，求得X和Y的最大值和最小值。

## [TODO9]实现getAccSize函数



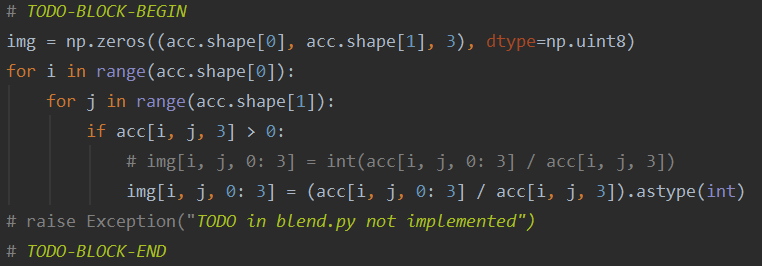
* 通过比较用于拼接全景图的每一幅图的X和Y的最大值和最小值，得出全景图的X和Y的最大值和最小值。

## [TODO10]实现accumulateBlend函数



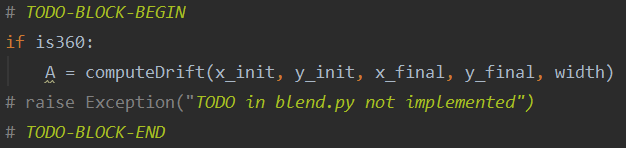
* 首先对用于拼接全景图的每一幅图计算在全景图中的边界。
* 再取全景图中边界中的每一个像素进行反向卷绕并对坐标进行归一化。
* 排除不在原图中的点和黑色像素点后对X方向的边界进行羽化。
* 对每个像素点各个通道的强度值和权重进行累加。

## [TOOD11]实现normalizeBlend函数



* 通过将每个像素点的每个通道的强度值除以累计的权重值进行归一化。
* 因为图像各个通道的强度值为整数，所以将除完的结果转化为int。
* 将处理过的各个通道的强度值赋值到新的矩阵从而去除权重信息。

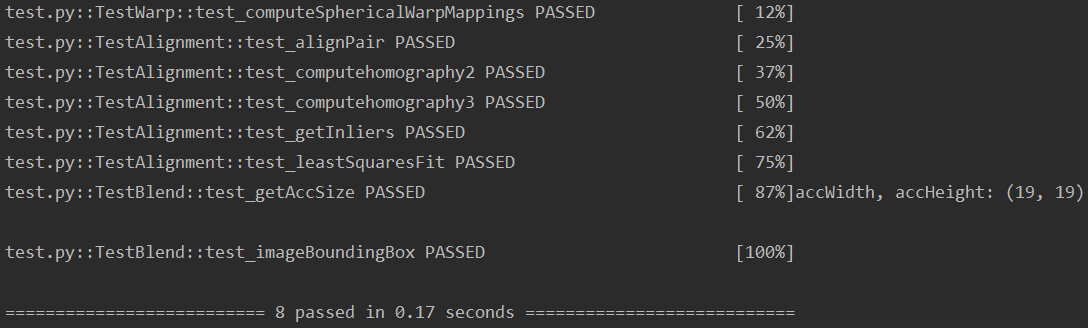
## [todo12]实现blendImages函数



* 判断is360参数来判断所生成的全景图是否为360°全景图，如果是360°全景图，调用已经写好的computeDrift函数消除漂移。

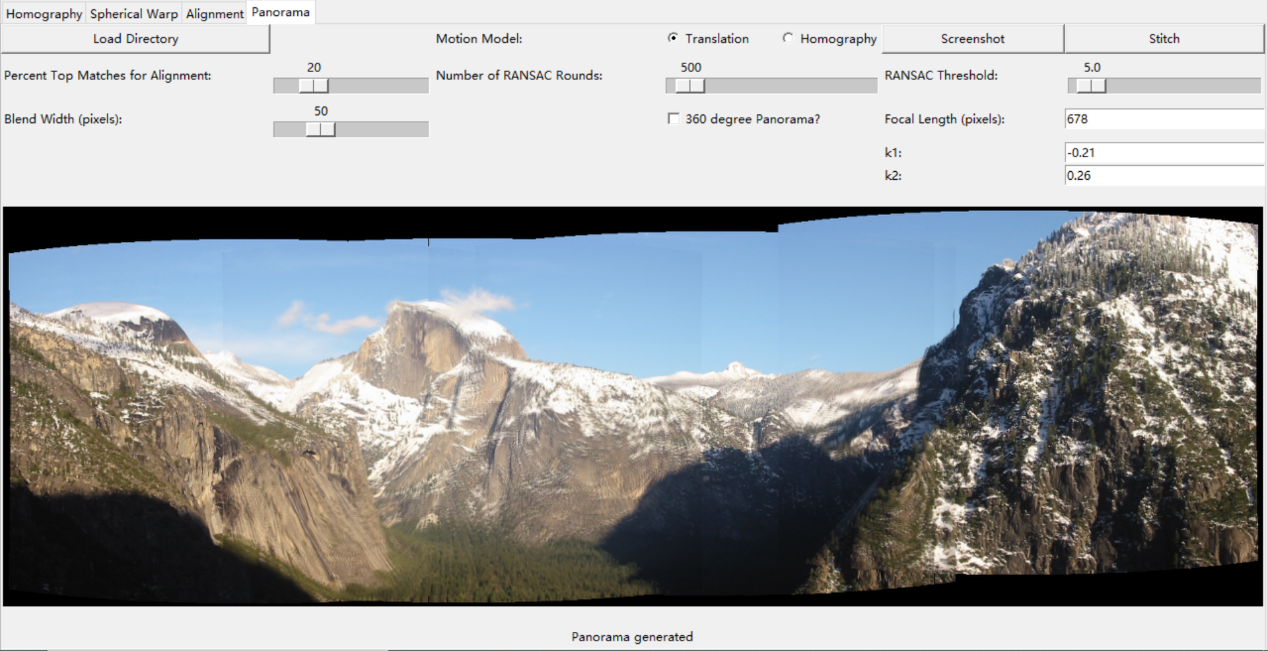
# 实验结果

## test.py测试结果

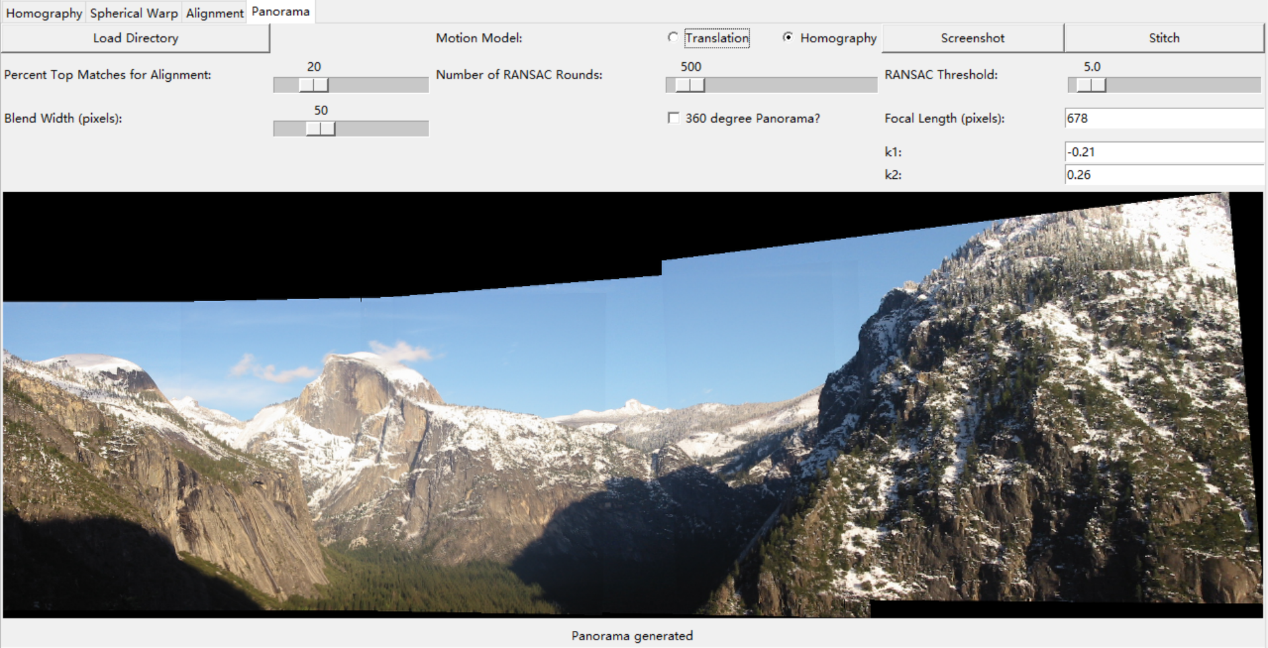


## 全景图（参数如图中所示）

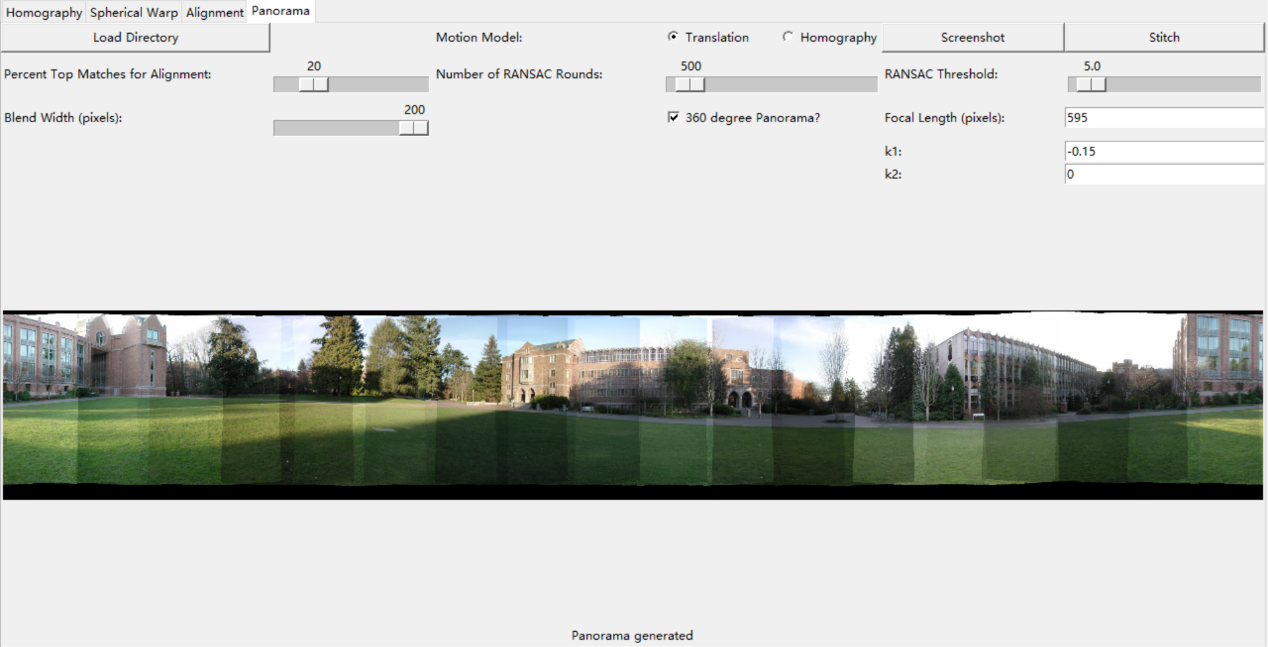
### 应用平移变换对 yosemite 数据集生成的全景图：



### 应用单应映射对 yosemite 数据集生成的全景图：

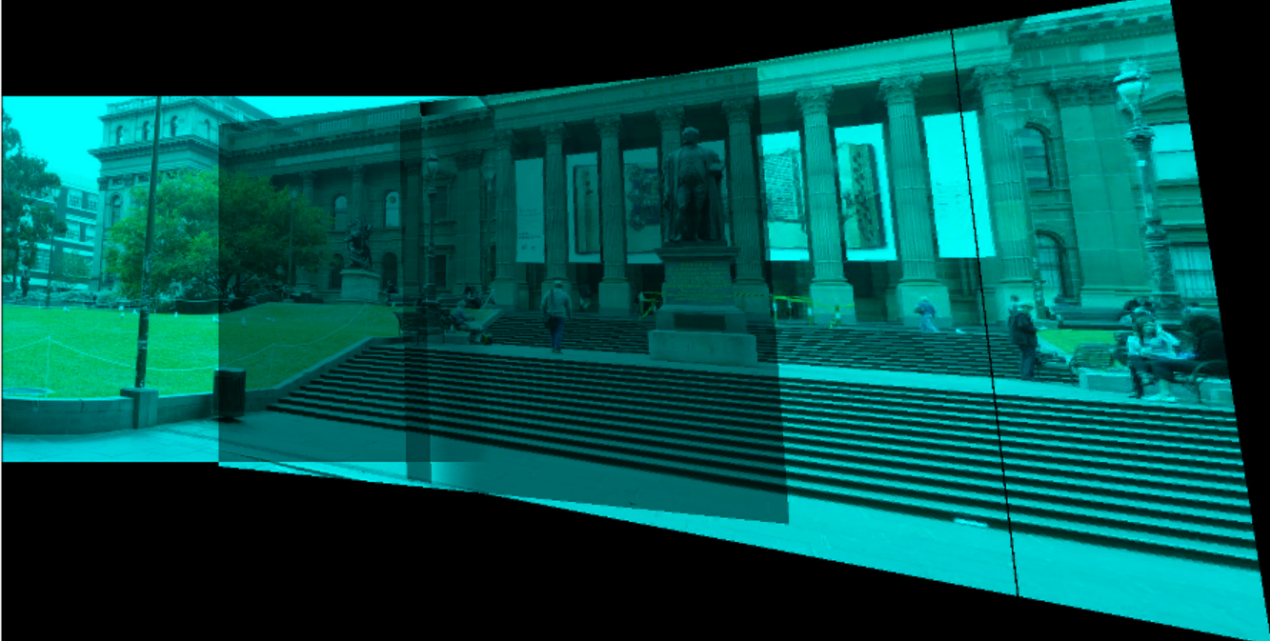


### 应用平移变换对 campus数据集生成的 360 度全景图：



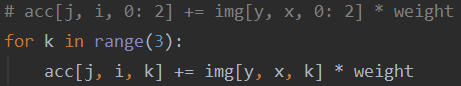
# 实验遇到的问题及心得

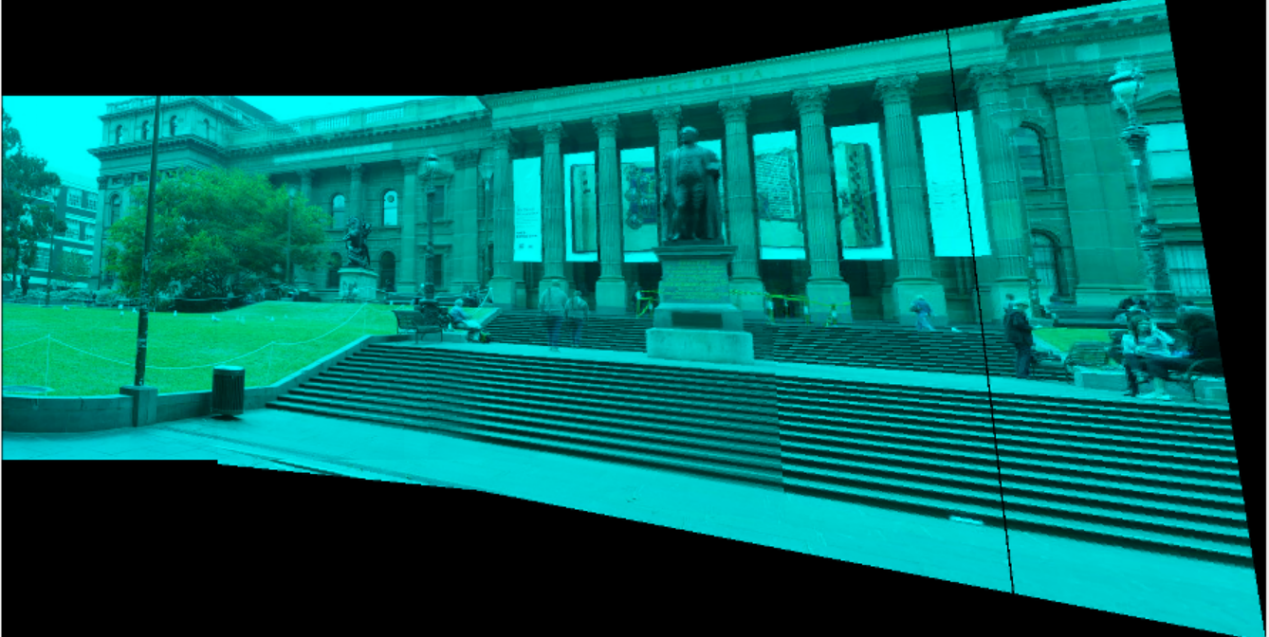
1. TODO10中羽化后的权值没有进行累计，导致合成的全景图如下：



改正后问题得以解决。

1. TODO10中给每个通道赋值时用切片操作（如下图注释掉的代码），导致导致合成的全景图如下：





改为循环后问题得已解决。

1. 本实验进行了多次将三维坐标归一化为二维坐标，一开始我都没有进行取整，导致报错，将所有归一化坐标转为int类型后还是会有报错，经过查找资料并与上机学长讨论，将所有归一化坐标统一上取整或下取整后问题得到解决。