# 实验报告-5

## 1 实验过程：

**双目立体匹配:**

立体匹配是一种技术，用来通过分析两张从不同角度拍摄的照片来创建场景的三维图像。想象一下你有两只眼睛，它们看到的景象稍有不同，但你的大脑能够将这些视角差异结合起来，让你感知到物体的深度。立体匹配就是计算机用来模仿这种过程的方法。

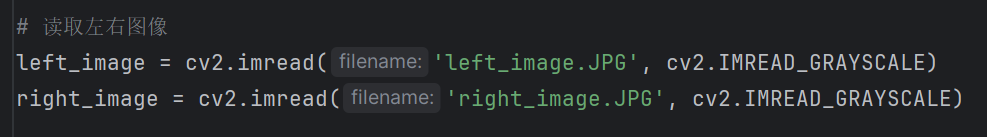
具体来说：

拍两张照片：用两个摄像头从稍微不同的位置拍摄同一个场景，就像人类的左眼和右眼。

找到相同点：计算机在这两张照片中寻找相同的点（比如树上的一个叶子或建筑物上的一个窗户）。

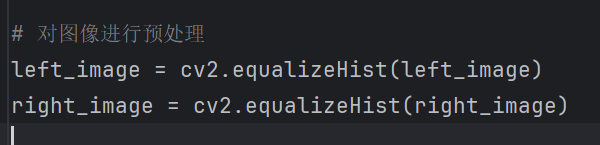
计算差异：通过比较这些点在两张照片中的位置差异，计算出这些点的深度信息。位置差异越大，物体离相机越近；差异越小，物体离相机越远。

首先读取左右图像，将他们转换为灰度图像：



然后对图像进行预处理：

对图像进行直方图均衡化处理，以增强图像的对比度。这有助于改善立体匹配的效果

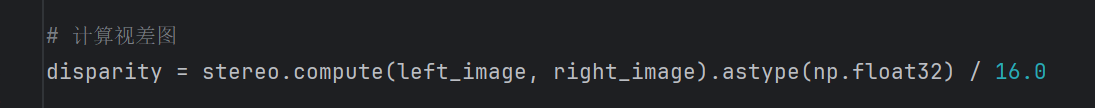


接下来创建一个StereoSGBM（Semi-Global Block Matching）对象，用于计算视差图。

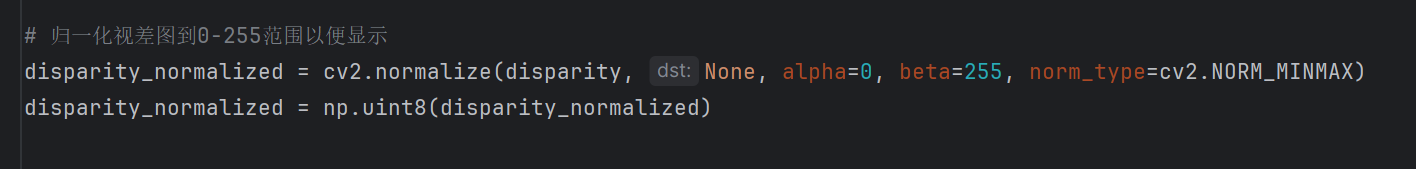
* minDisparity: 视差搜索的最小值（从零开始）。
* numDisparities: 视差搜索范围，必须是16的倍数。
* blockSize: 匹配块的大小，通常为奇数。
* P1: 控制视差平滑度的参数，与图像梯度相关。
* P2: 控制视差平滑度的参数，与图像内容相关。
* disp12MaxDiff: 左右视差检查的最大允许差异。
* uniquenessRatio: 用于视差唯一性验证的百分比。
* speckleWindowSize: 视差连通区域的最大窗口大小。
* speckleRange: 视差变化的最大范围。



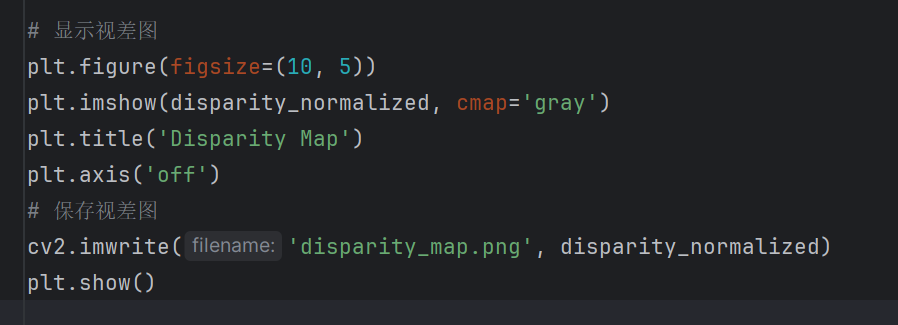
使用stereo.compute方法计算视差图，并将其转换为浮点型，然后除以16得到实际视差值。



使用cv2.normalize函数将视差图归一化到0-255的范围，以便于显示。然后将其转换为8位无符号整数类型。



然后将视差图展示、保存图片。



## 2 结果分析：

左右图像为：



得到的视差图为：



生成的视差图反映了左右图像之间的深度差异。以下是对视差图的具体分析：

视差图特征：

较亮的区域表示较小的视差，即离相机较近的物体。

较暗的区域表示较大的视差，即离相机较远的物体。

图像细节：

在视差图中，可以看到立体匹配算法较好地捕捉到了物体的轮廓和深度变化。特别是预处理后的图像对比度增强，使得匹配点更为明显。