**图像视差匹配**

**一、实验目的**

本实验旨在通过立体匹配算法计算图像的视差图，并分析不同参数对匹配效果的影响，从而理解立体视觉的基本原理以及立体匹配算法的实现过程。

**二、实验原理**

立体匹配是通过比较左右两幅图像中对应像素点的差异来计算其视差，从而推断物体的深度信息。立体匹配的基本原理是利用了人类视觉系统中的双眼视差原理，即左右眼观察到的同一物体在左右视网膜上的位置差异来判断距离远近。

在立体匹配过程中，我们首先将左右图像进行配准，使得它们的特征点对应。然后通过比较左右图像中对应像素点的灰度值或特征信息来计算它们之间的视差。最终得到的视差图表示了图像中每个像素点的深度信息。

常见的立体匹配方法，包括基于区域的方法，基于特征的方法，深度学习方法。本实验所用的SGBM算法是一种基于区域的立体匹配算法，它通过优化代价聚合的方式，综合考虑了全局和局部的信息，从而得到更准确的视差图。SGBM算法的主要步骤包括：

代价计算：计算左右图像中对应像素点的匹配代价。代价可以使用像素间的灰度差异或特征间的距离等。

代价聚合：将每个像素点的代价与其邻域内像素点的代价相结合，得到每个像素点的最小代价。

路径优化：通过动态规划或其他优化方法，计算出全局的最小代价路径。

视差计算：根据最小代价路径，得到每个像素点的视差值。

**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。使用 cv2.imread 函数将图像文件加载到内存中。

3.计算视差图。使用StereoSGBM\_create创建一个SGBM立体匹配对象，SGBM的参数有：minDisparity、numDisparities、blockSize、P1、P2、disp12MaxDiff、uniquenessRatio、speckleWindowSize、speckleRange。在实验中，调节numDisparities、blockSize的大小，分析不同参数对匹配效果的影响。利用compute计算视差图。

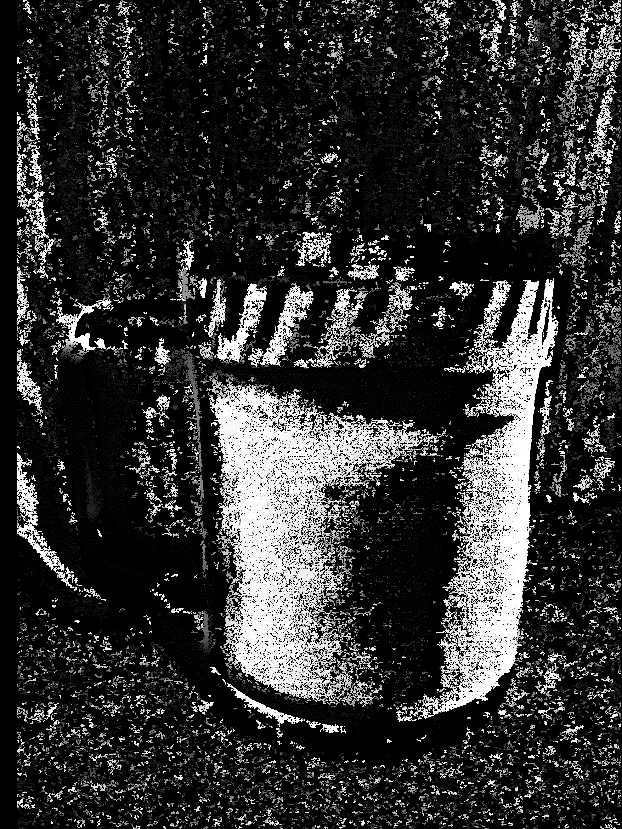
4.显示视差图并保存结果。

**四、实验结果**

左 右

调整参数：视差范围与匹配块大小。视差图a两参数分别为16\*5、5，视差图b两参数分别为16\*5、7，视差图c两参数分别为16\*3、5。

a b



c

**五、结果分析**

在本实验中，我使用了SGBM算法来计算视差图，并通过调整不同参数来观察它们对视差图效果的影响。numDisparities 指定了视差搜索的范围, 较大的视差范围可以处理更大的深度变化，但可能引入更多噪声；使用较小的视差范围，计算更快，细节部分更准确，但在处理深度变化大的区域时可能出现视差不足的情况。blockSize 是匹配块的大小。较大的块大小可以平滑视差图，减少噪声，但可能会模糊细节；较小的块大小可以保留更多的细节，但可能会引入更多的噪声。