

**《计算机视觉工程实践》报告**

学号：823104010011

姓名：高双猛

**图像视差匹配**

**实验要求**

通过立体匹配得到两张图片的视差图。

**实验原理**

图像视差匹配是计算机视觉中用于获取场景深度信息的一种技术。它基于立体视觉原理，通过分析左右两个摄像头捕获的图像之间的差异来估计场景中物体的距离。在本实验中，我们使用了OpenCV库中的StereoSGBM（Semi-Global Block Matching）算法来计算视差图。

StereoSGBM算法是一种高效的视差匹配算法，它通过在图像块之间进行匹配来估计视差。算法的关键在于选择合适的参数，包括最小视差（minDisparity）、视差级数（numDisparities）和块大小（blockSize）。这些参数会影响算法的性能和精度。

**实验步骤**

导入必要的库：导入OpenCV库用于图像处理和视差计算，以及Matplotlib库用于图像显示。

定义计算视差图的函数：创建一个名为compute\_disparity\_map的函数，该函数接受左右视图图像作为输入，并返回计算得到的视差图。

创建StereoSGBM对象：在compute\_disparity\_map函数中，使用StereoSGBM算法创建一个StereoSGBM对象，并设置其参数。

计算视差图：使用StereoSGBM对象的compute方法计算左右视图图像之间的视差图。

视差图归一化：将计算得到的视差图进行归一化处理，以便更好地显示和分析。

定义显示视差图的函数：创建一个名为display\_disparity\_map的函数，用于显示左右视图图像和视差图。

创建Matplotlib子图：在display\_disparity\_map函数中，创建一个包含三个子图的Matplotlib图形，用于显示左右视图图像和视差图。

显示图像和视差图：使用imshow方法在子图中显示图像，并使用colorbar方法为视差图添加颜色条。

主函数：在main函数中，读取左右视图图像，调用compute\_disparity\_map函数计算视差图，并调用display\_disparity\_map函数显示图像和视差图。

**实验结果**

** **

图1 输入图片

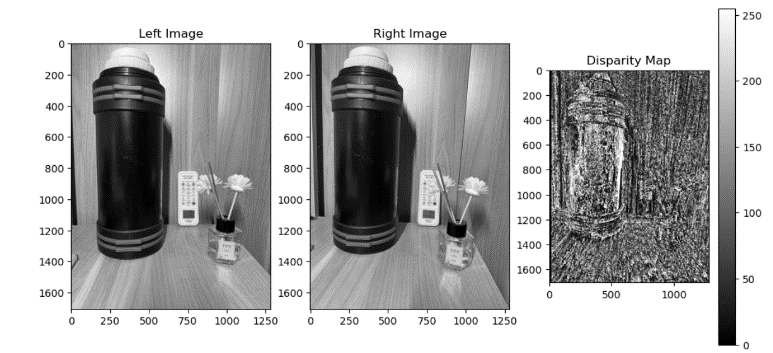


图2 实验结果

**实验总结**

通过本次实验，我成功地实现了图像视差匹配的基本流程。我使用了OpenCV库中的StereoSGBM算法来计算视差图，并使用Matplotlib库来显示图像和视差图。实验过程中，我了解到了视差图的重要性，它可以帮助我们估计场景中物体的距离，对于三维重建和机器人导航等领域具有重要意义。实验中，我注意到选择合适的StereoSGBM算法参数对于获得高质量的视差图至关重要。此外，通过Matplotlib的可视化，我可以直观地看到视差图与原始图像之间的关系，这有助于我们更好地理解立体视觉的原理。在未来的工作中，可以进一步探索如何优化算法参数，以提高视差匹配的精度和效率。同时，也可以尝试使用其他视差匹配算法，以比较它们的性能和适用场景。