# 计算机视觉实践-练习5实验报告

目录

[计算机视觉实践-练习5实验报告 1](#_Toc134027674)

[一、 实验目的 1](#_Toc134027675)

[二、 实验原理 1](#_Toc134027676)

[三、 实验步骤 2](#_Toc134027677)

[四、 数据集 3](#_Toc134027678)

[五、 代码程序 4](#_Toc134027679)

[六、 实验结果 5](#_Toc134027680)

[七、 实验分析与总结 5](#_Toc134027681)

1. 实验目的

* 图像视差匹配，通过立体匹配得到两张图像的视差图。

1. 实验原理

立体匹配算法也称为视差算法,目的是求解两幅图片之间的视差值。主要是通过找出每对图像间的对应关系，根据三角测量原理，得到视差图；在获得了视差信息后，根据投影模型很容易地可以得到原始图像的深度信息和三维信息。这里采用归一化积相关算法（NCC）算法。

匹配流程如下：

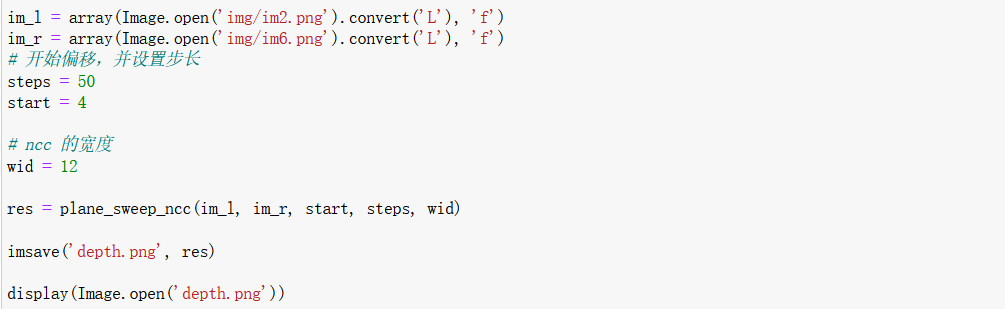
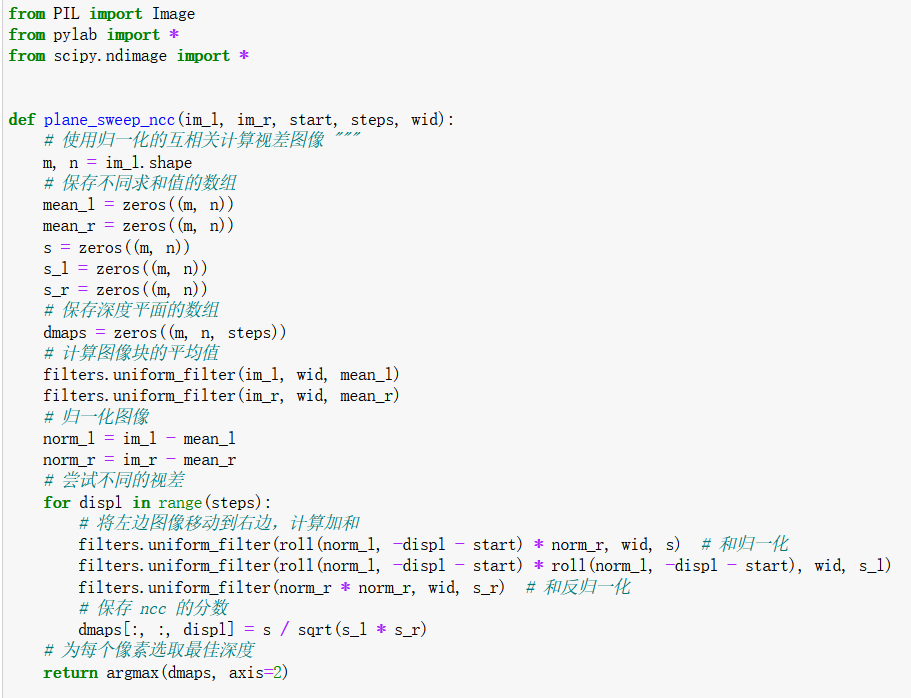
1. 采集图像：通过标定好的双目相机采集图像，当然也可以用两个单目相机来组合成双目相机。
2. 极线校正：校正的目的是使两帧图像极线处于水平方向，或者说是使两帧图像的光心处于同一水平线上。通过校正极线可以方便后续的NCC操作。
3. 特征匹配：这里便是我们利用NCC做匹配的步骤啦，匹配方法如上所述，右视图中与左视图待测像素同一水平线上相关性最高的即为最优匹配。完成匹配后，我们需要记录其视差d，即待测像素水平方向xl​与匹配像素水平方向xr​之间的差值d=xr​−xl​ ，最终我们可以得到一个与原始图像尺寸相同的视差图D。
4. 深度恢复：通过上述匹配结果得到的视差图D，我们可以很简单的利用相似三角形反推出以左视图为参考系的深度图。
5. 实验步骤
6. **匹配代价计算：匹配代价计算的目的是衡量待匹配像素与候选像素之间的相关性。两个像素无论是否为同名点，都可以通过匹配代价函数计算匹配代价，代价越小则说明相关性越大，是同名点的概率也越大。匹配代价的算法由很多种，本文采用NCC算法匹配。**
7. **代价聚合：代价聚合的根本目的是让代价值能够准确的反映像素之间的相关性。上一步匹配代价的计算往往只会考虑局部信息，通过两个像素邻域内一定大小的窗口内的像素信息来计算代价值，这很容易受到影像噪声的影响，而且当影像处于弱纹理或重复纹理区域，这个代价值极有可能无法准确的反映像素之间的相关性，直接表现就是真实同名点的代价值非最小。**
8. **视差获取：对于区域算法来说，在完成匹配代价的叠加以后，视差的获取就很容易了，只需在一定范围内选取叠加匹配代价最优的点（SAD和SSD取最小值，NCC取最大值）作为对应匹配点，如胜者为王算法WTA（Winner-take-all）。而全局算法则直接对原始匹配代价进行处理，一般会先给出一个能量评价函数，然后通过不同的优化算法来求得能量的最小值，同时每个点的视差值也就计算出来了。**
9. **视差细化：大多数立体匹配算法计算出来的视差都是一些离散的特定整数值，可满足一般应用的精度要求。但在一些精度要求比较高的场合，如精确的三维重构中，就需要在初始视差获取后采用一些措施对视差进行细化，如匹配代价的曲线拟合、图像滤波、图像分割等。**
10. 数据集

采用下面两张图片作为测试图片如下：

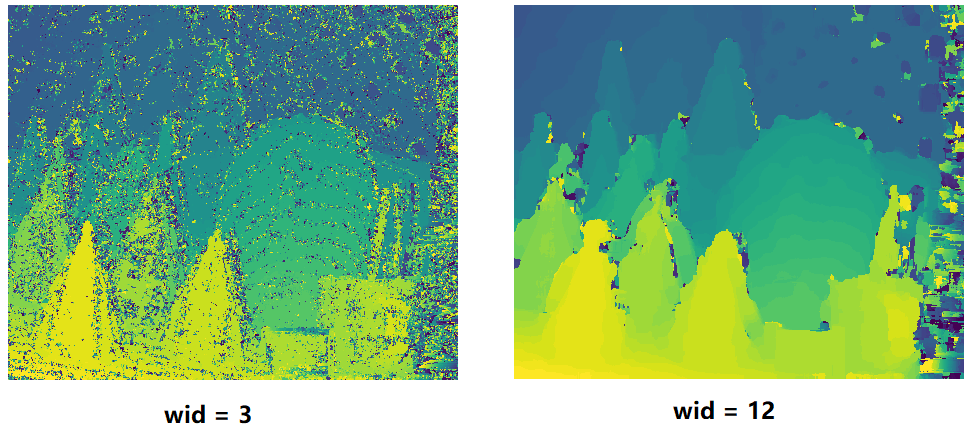


1. 代码程序

程序代码如下：



1. 实验结果



对比可见当窗口值过低时低纹理区域容易出现误匹配，匹配精度较低，并且噪声也很多。随着窗口值的增大，其中错误的匹配的区域也随之慢慢被矫正，但是在信息的细节上越来越模糊。

1. 实验分析与总结

由上诉实验可知，wid值越小图片对噪声越敏感，在低纹理区会出现很多错误匹配，但细节鲁棒好，值越大噪声鲁棒好，错误匹配随之被纠正但细节随之会被忽略掉，导致图片的精度小。