

# Homework 4

林宇辰 2200013211

December 2023

## 1 Simple Stereo System

遍历可能的差距值，对两张图片的 window 求相似度，找出每个像素最匹配的对应像素，求出 map。可视化效果见图 1。可以发现，我求出的深度图对比真实的深度图有异同之处。首先，从整体上看，我的深度图能够较好地刻画图片的整体深度信息。其次，我的深度图在精细处的深度计算效果不好，例如灯的支架部分，这可能是由于差异函数误差的问题；同时，我的深度图求出的绝对深度与 groundtruth 是有差别的（从画面的明暗程度来看）。

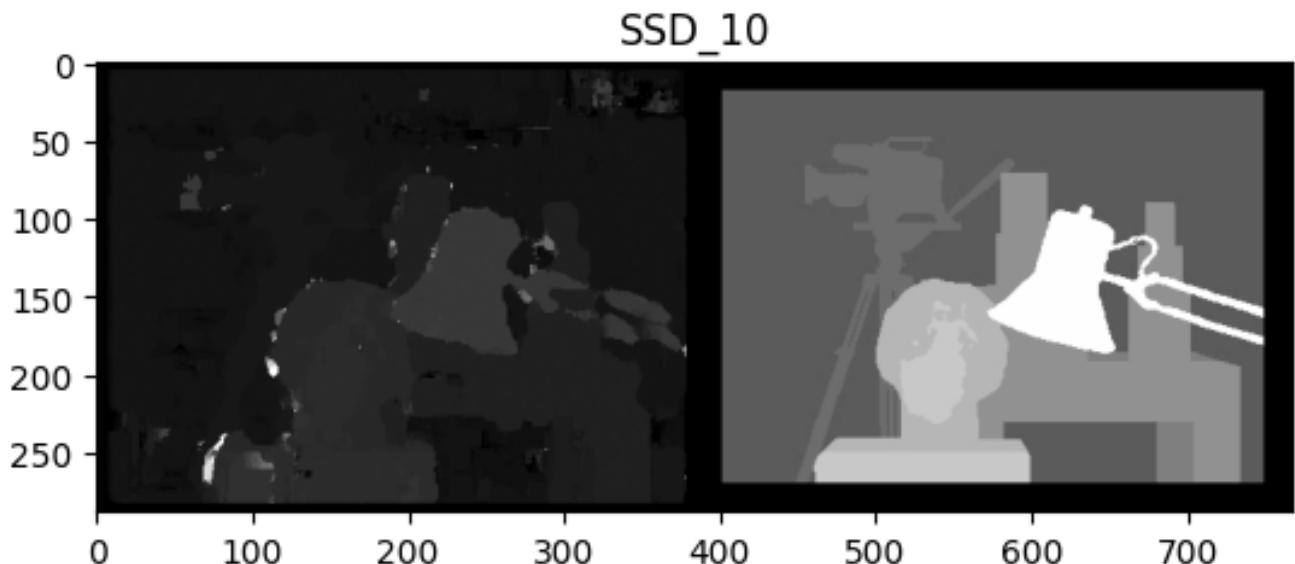


图 1: Simple Stereo System

## 2 Algorithm Settings

对比图如下。

### 2.1 Window size

小的 window size 能够更精准地采集目标区域，但是大的 window size 能够捕捉到更多的信息。从时间上看，用同样的 SSD 函数，窗口大小为 2、10、20 分别需要 40s、45s、46s。其实时间差距不大，大的窗口会减少一些显示 for 循环，但是在内部的矩阵运算中需要计算更大的矩阵。

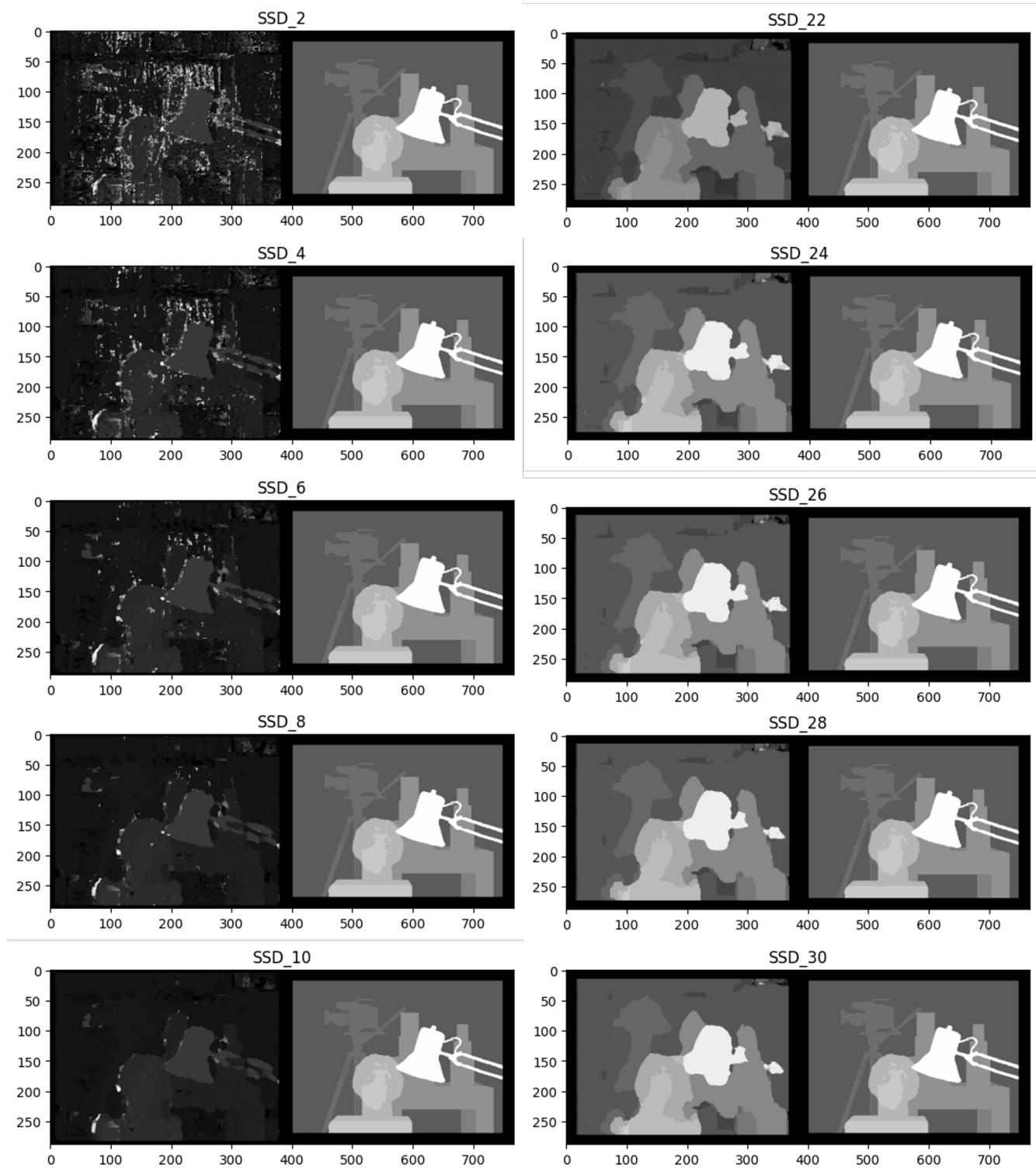


图 2: SSD



图 3: SAD

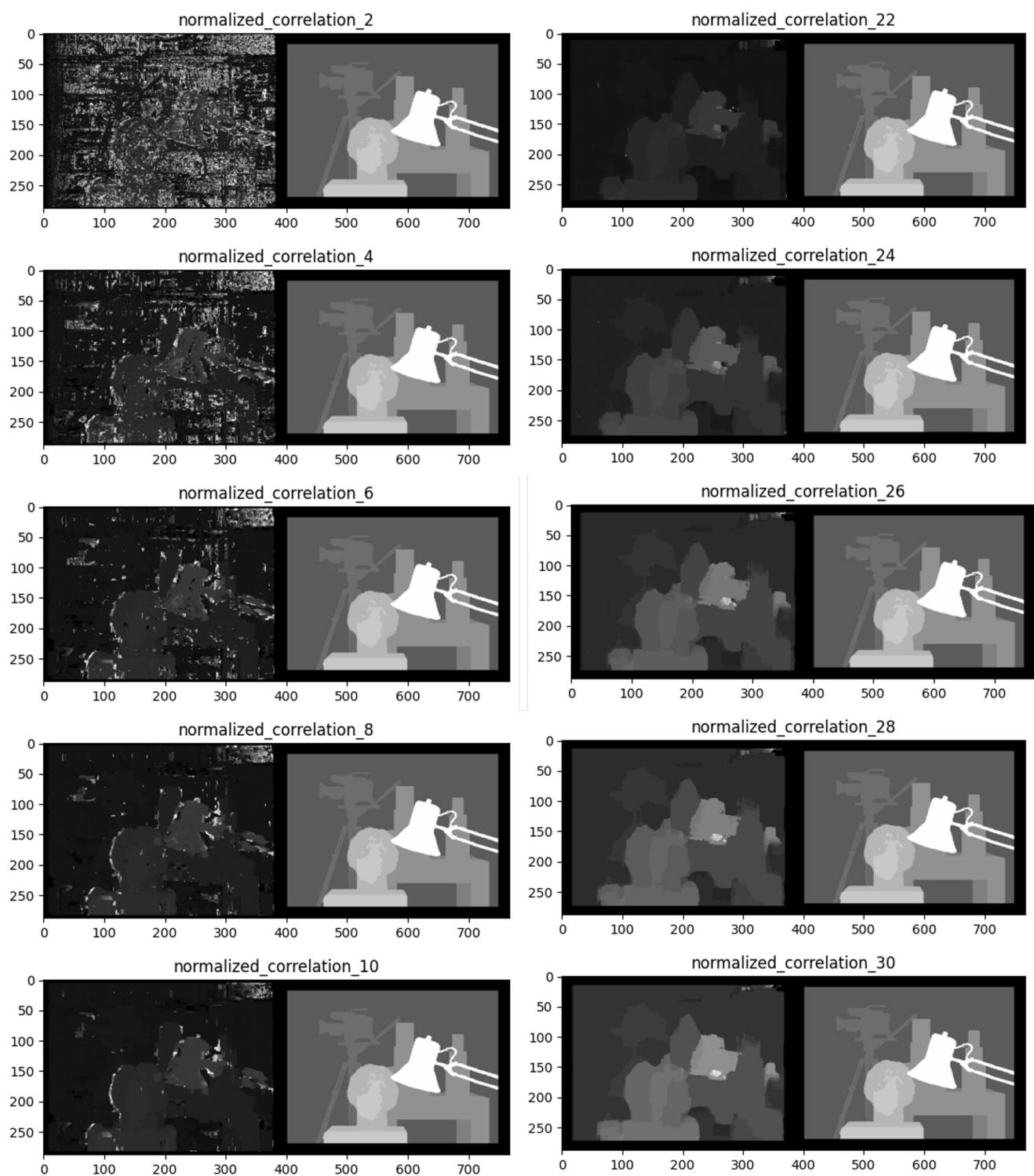


图 4: normalized correlation

## 2.2 Disparity range

0 到 64 是课程 slides 中合理的差距，我也沿用了 64 的最大差距。对于搜索的方式，直接按照线性的搜索方式从原像素出发往左比较就行。

## 2.3 Matching function

SAD 的效果最差，他不能很好地控制异常值。SAD 和 normalized correlation 的效果差不多，在小窗口时 SAD 表现的更好。从时间上看，SAD 和 SSD 的时间差不多，normalized correlation 特别慢。

## 2.4 Final settings

选用 window size = 10, max disparity = 64, SSD 即可。效果如图 5。可以发现和 groundtruth 差别不大。

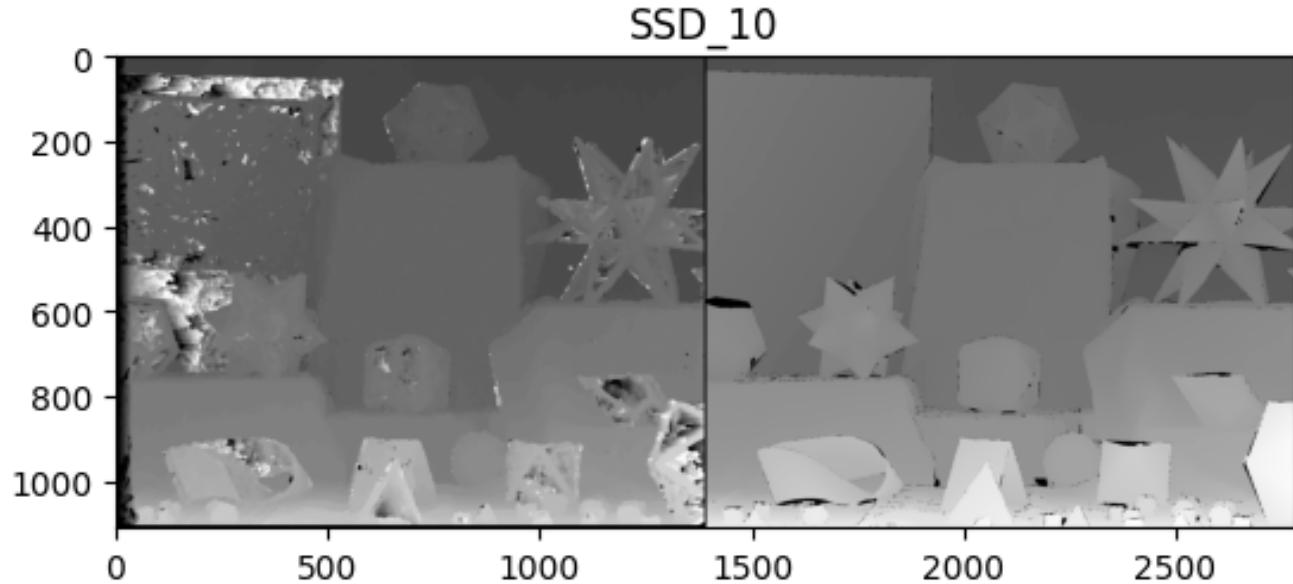


图 5: my settings

## 3 Visualize the depth map in 3D

计算深度。这里为了美观，我直接抛弃了差距过小的点对（会出现深度特别大的情况），并且为了体现出深度的差异，需要调大分母的常数。可视化如图 6所示。

## 4 Dynamic Programming Algorithm

按照课件，对于每一行上的像素，首先建立一个  $\text{cols} \times \text{cols}$  的矩阵，计算像素与另一幅图中位于其左侧阈值内的像素的代价。然后使用动态规划，规划出最优路径。最后，使用 full Occlusion，填补未匹配的像素。可视化效果如图 7所示。

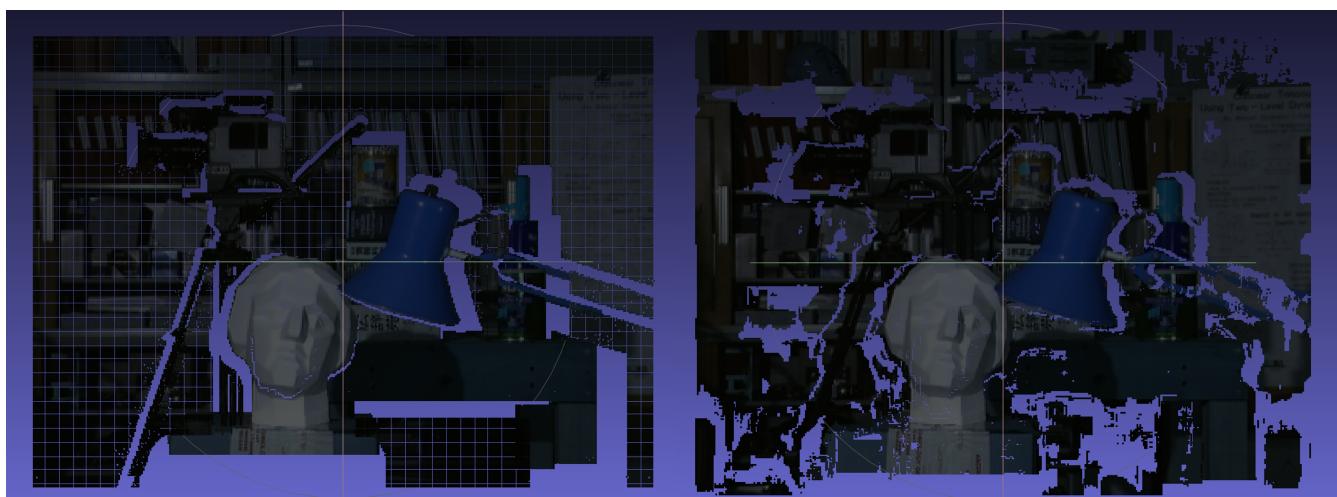


图 6: 3D map

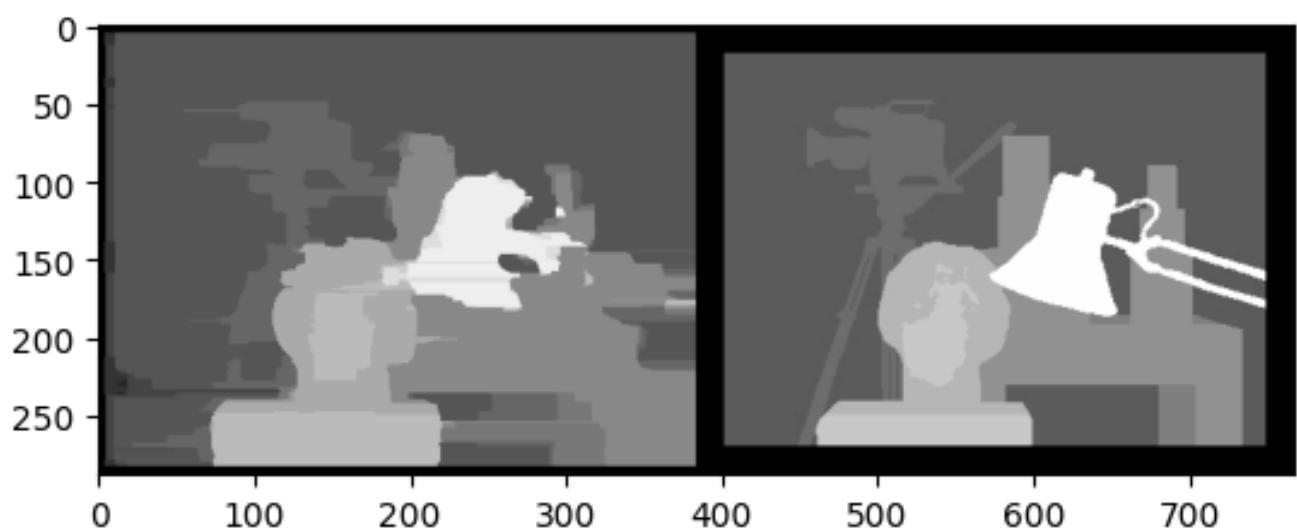


图 7: Dynamic Programming Algorithm