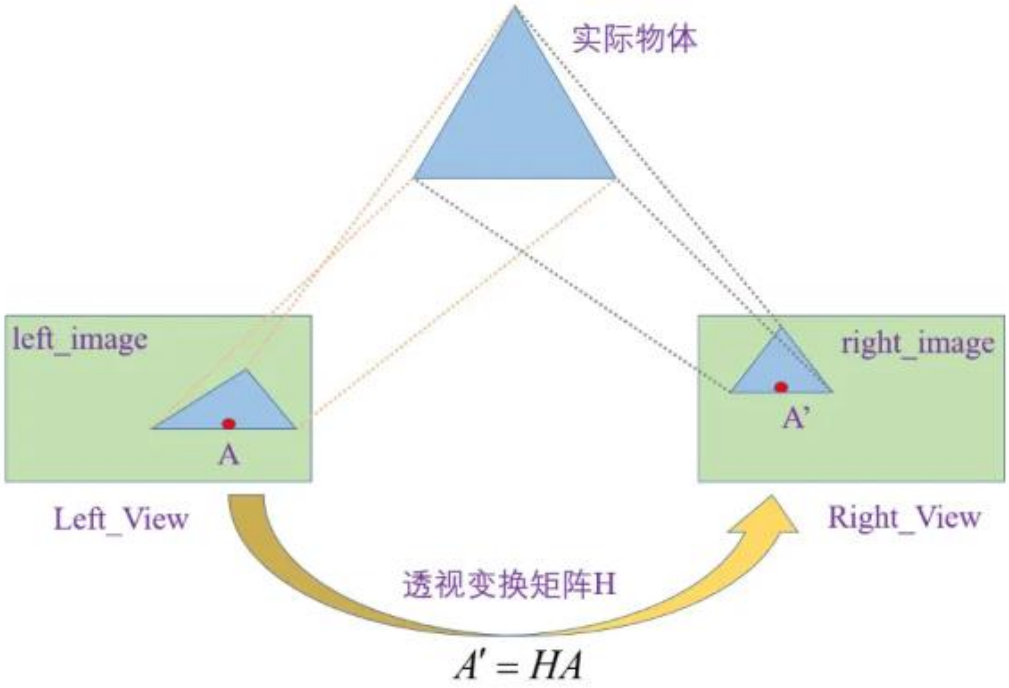
计算机视觉实践报告

目录

一、单应性变换

1.实验原理

用无镜头畸变的相机从不同位置拍摄同一平面物体的图像之间存在单应性，可以用透视变换矩阵来表示图像之间的映射关系。如下图，将相机分别放到左右两边拍摄同一个三角形，可获得两张图像 left_image 和 right_image，left_image 上的点可以通过透视变换到 right_image 图像上对应的位置。



假设 Left_View 图像上的点为(xi,yi)，Right_View 图像上的点为(xi',yi')，则透视变换过程如下：

$$\begin{bmatrix} x_i' \\ y_i' \\ 1 \end{bmatrix} = H_{3\times3}\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11}x_i + h_{12}y_i + h_{13} \\ h_{21}x_i + h_{22}y_i + h_{23} \\ h_{31}x_i + h_{32}y_i + h_{33} \end{bmatrix}$$

$$\therefore \quad x_i' = \frac{h_{11}x_i + h_{12}y_i + h_{13}}{h_{31}x_i + h_{32}y_i + h_{33}}$$

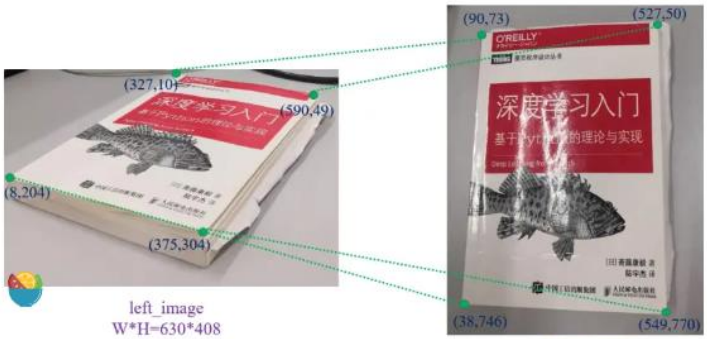
$$y_i' = \frac{h_{21}x_i + h_{22}y_i + h_{23}}{h_{31}x_i + h_{32}y_i + h_{33}}$$

可见，一组匹配点(xi,yi)-(xi',yi')可以得到 2 组方程。

在求矩阵 H 时一般添加约束 h33=1(如下图)，所以单应性矩阵 H 虽然有 9 个未知数，实际却只有 8 个未知数，即 h'11~h'32；由于一组匹配点可以得到 2 组方程，所以要确定 8 个未知数需要 4 组不共线的匹配点，即可求解矩阵 H 的唯一解。

$$H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} = \begin{bmatrix} \frac{h_{11}}{h_{33}} & \frac{h_{12}}{h_{33}} & \frac{h_{13}}{h_{33}} \\ \frac{h_{21}}{h_{33}} & \frac{h_{22}}{h_{33}} & \frac{h_{23}}{h_{33}} \\ \frac{h_{31}}{h_{33}} & \frac{h_{32}}{h_{33}} & 1 \end{bmatrix} = \begin{bmatrix} h_{11}' & h_{12}' & h_{13}' \\ h_{21}' & h_{22}' & h_{23}' \\ h_{31}' & h_{32}' & 1 \end{bmatrix}$$

如下图，我们分别从两个视角拍摄同一本书，得到对应的两张图片，只需找出两张图片的 4 组匹配点，就可以确定出两张图片之间的单应性矩阵：



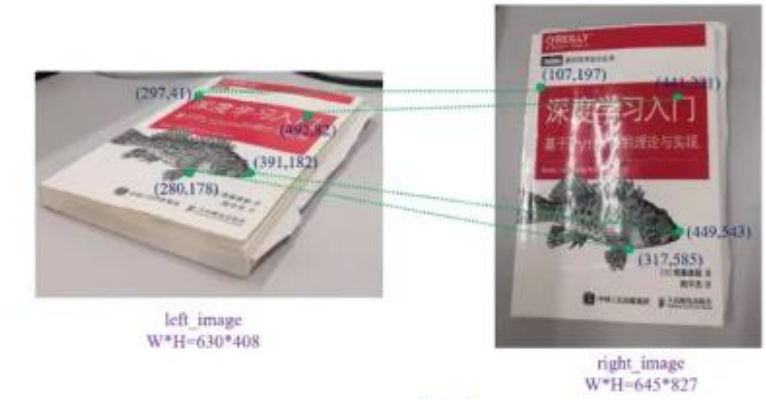
2.实验结果分析

结果如下

通过单应性变换将 left 空间的图片变换到 right 空间，warp 为变换的结果，warp 图片与 right 图片处于同一空间，像素点一一对应：

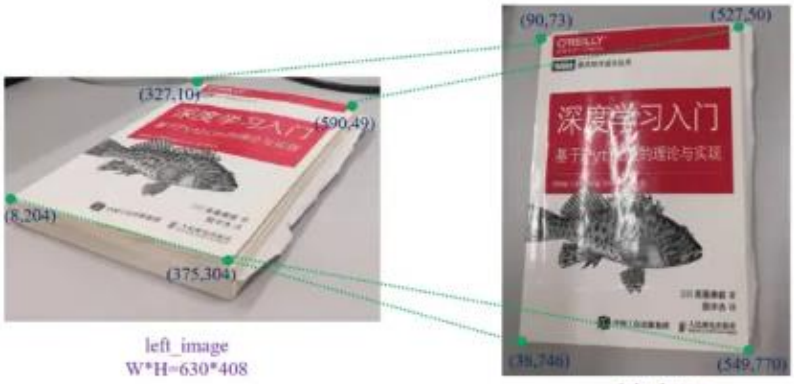
注意：

选择的匹配点不同，计算出的单应性矩阵也会有稍许差异，如下图：



单应性矩阵H

```
[[ 1.44590198e+00  2.32789420e+00 -4.12219944e+02]
 [-6.22414348e-01  3.89115828e+00  2.32734071e+02]
 [-5.38838227e-05  1.67973108e-03  1.00000000e+00]]
```



单应性矩阵H

```
[[ 1.44183886e+00  2.16282553e+00 -4.02935816e+02]
 [-6.19023115e-01  3.64610189e+00  2.39100467e+02]
 [-4.07556192e-05  1.52576024e-03  1.00000000e+00]]
```

二、立体匹配

1.实验目的

图像视差匹配，通过立体匹配（Stereo Matching）得到两张图像的视差图需要详细的实验过程和结果分析。

2.实验原理

立体匹配也称作视差估计（disparity estimation），或者双目深度估计。其输入是一对在同一时刻捕捉到的，经过极线校正的左右图像。而它的输出是由参考图像（一般以左图作为参考图像）中每个像素对应的视差值所构成的视差图d。视差是三维场景中某一点在左右图像中对应点位置的像素级差距。如下图：

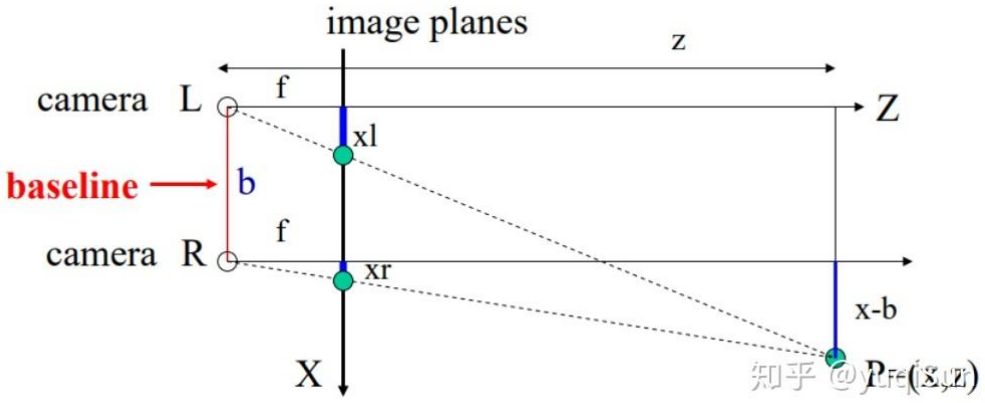


图1

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。每个像素在搜索同名点之前，往往会指定一个视差搜索范围D,视

差搜索时将范围限定在D内，用一个大小为W×H×D（W为影像宽度，H为影像高度）的三维矩阵C来存储每个像素在视差范围内每个视差下的匹配代价值。矩阵C通常称为DSI。

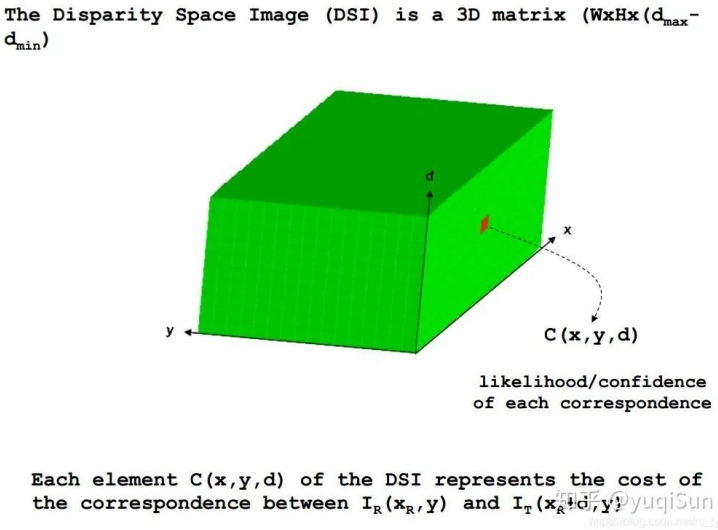


图2 DSI示意图

代价聚合类似于一种视差传播步骤，信噪比高的区域匹配效果好，初始代价能够很好的反映相关性，可以更准确的得到最优视差值，通过代价聚合传播至信噪比低、匹配效果不好的区域，最终使所有影像的代价值都能够准确反映真实相关性。常用的代价聚合方法有扫描线法、动态规划法、SGM算法中的路径聚合法等。

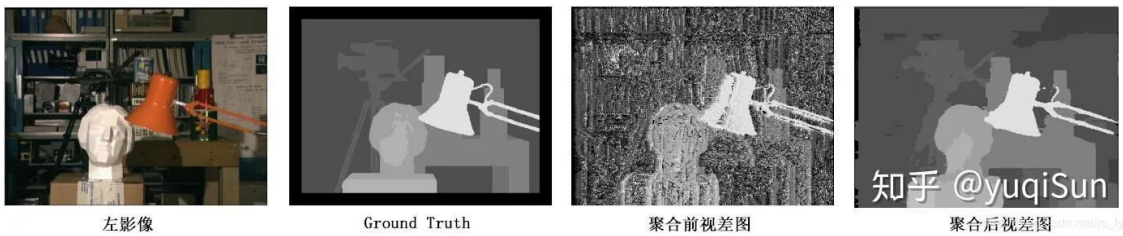


图3

视差计算即通过代价聚合之后的代价矩阵S来确定每个像素的最优视差值，通常使用赢家通吃算法（WTA，Winner-Takes-All）

来计算，如图2所示，即某个像素的所有视差下的代价值中，选择最小代价值所对应的视差作为最优视差。这一步非常简单，这意味着聚合代价矩阵S的值必须能够准确的反映像素之间的相关性，也表明上一步代价聚合步骤是立体匹配中极为关键的步骤，直接决定了算法的准确性。

视差优化的目的是对上一步得到的视差图进行进一步优化，改善视差图的质量，包括剔除错误视差、适当平滑以及子像素精度优化等步骤，一般采用左右一致性检查（Left-Right Check）算法剔除因为遮挡和噪声而导致的错误视差；采用剔除小连通区域算法来剔除孤立异常点；采用中值滤波（Median Filter）、双边滤波（Bilateral Filter）等平滑算法对视差图进行平滑；另外还有一些有效提高视差图质量的方法如鲁棒平面拟合（Robust Plane Fitting）、亮度一致性约束（Intensity Consistent）、局部一致性约束（Locally Consistent）等也常被使用。

3.实验步骤

1、误差能量函数

选取匹配计算区域窗口大小是(m*n),d是视差，我们需要先定一个视差搜寻范围如dmax=40。

$$e(i,j,d)=\frac{1}{3\cdot n\cdot m}\cdot\sum_{x=i}^{i+n}\sum_{y=j}^{j+m}\sum_{k=1}^{3}\left(L(x,y+d,k)-R(x,y,k)\right)^2$$

由于该算法对噪声敏感，进一步计算平均error energy。

$$\tilde{e}(i,j,d)=\frac{1}{n\cdot m}\sum_{x=i}^{i+n}\sum_{y=j}^{j+m}e(x,y,d)$$

2、基于最小平均误差能量的视差图

选取error energy最小的d作为视差图中（i,j）点的d，得到视差图

3、计算可靠度，生成具有可靠视差的视差图。

4.实验结果

图4

图5

图6

5.实验分析与总结

初始图像数据集的选择对于结果的准确率有直接的影响，同时通过对比发现，SGBM 算法的表现要远优于 BM 算法，因此采用 SGBM 算法获取视差图