

计算机视觉 课程实验报告

学号：201900130151

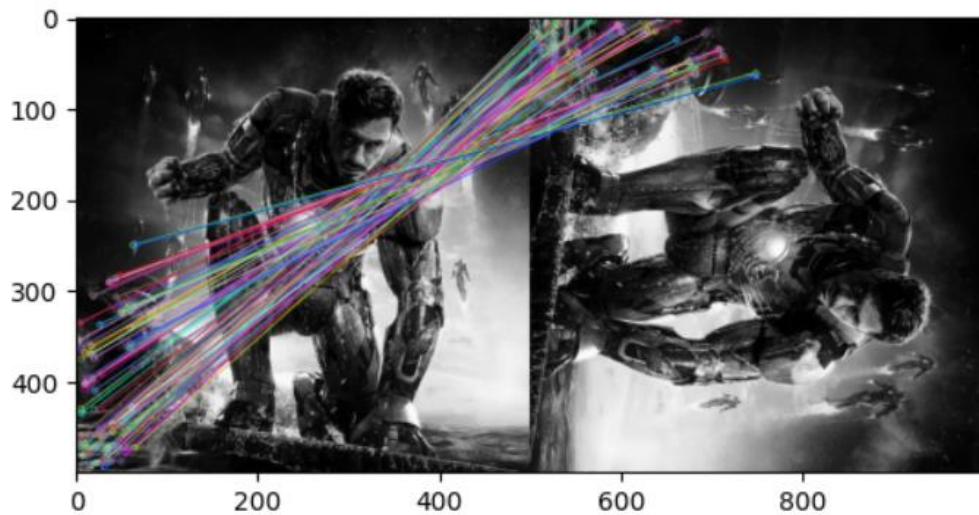
姓名：莫甫龙

实验题目：基于学习的局部特征

实验过程中遇到和解决的问题：

（记录实验过程中遇到的问题，以及解决过程和实验结果。可以适当配以关键代码辅助说明，但不要大段贴代码。）

首先使用 sift，得到的结果如下：



首先运行 r2d2，然后得到图片的关键点坐标和特征向量，然后取出两张图的关键点坐标和特征向量。

```
import numpy as np
import cv2 as cv

l=np.load('C:/Users/15339/Desktop/计算机视觉/实验/实验十一/lab/r2d2-master/imgs/hero.jpg.r2d2')
p=np.load('C:/Users/15339/Desktop/计算机视觉/实验/实验十一/lab/r2d2-master/imgs/hero2.jpg.r2d2')
kp1=[]
for i in range(300):
    kp1.append([l["keypoints"][i][0],l["keypoints"][i][1]])
kp1=np.array(kp1)
des1=[]
for i in range(300):
    des1.append(l["descriptors"][i])
des1=np.array(des1)

kp2=[]
for i in range(300):
    kp2.append([p["keypoints"][i][0],p["keypoints"][i][1]])
kp2=np.array(kp2)
des2=[]
for i in range(300):
    des2.append(p["descriptors"][i])
des2=np.array(des2)
```

接着进行特征点匹配，因为匹配的点比较多，所以再筛选掉一部分，然后将两张图拼接起来，然后再在上面画出匹配的关键点。

```
import matplotlib.pyplot as plt

img1 = cv.imread("C:\\Users\\15339\\Desktop\\pictures\\hero.jpg")
img2 = cv.imread("C:\\Users\\15339\\Desktop\\pictures\\hero2.jpg")
matcher = cv.DescriptorMatcher_create('BruteForce')
matches = matcher.knnMatch(des1, des2, 2)
good = []
for m in matches:
    if len(m) == 2 and m[0].distance <= m[1].distance:
        good.append((m[0].trainIdx, m[0].queryIdx))

(hA,wA) = img1.shape[:2]
(hB,wB) = img2.shape[:2]
vis = np.zeros((max(hA,hB), wA+wB, 3), dtype='uint8')
vis[0:hA, 0:wA] = img1
vis[0:hB, wA:] = img2
for (queryIdx,trainIdx) in good:
    ptA = (int(kp1[queryIdx][0]), int(kp1[queryIdx][1]))
    ptB = (int(kp2[trainIdx][0])+wA, int(kp2[trainIdx][1]))
    cv.line(vis, ptA, ptB, (0, 255, 0), 1)
cv.imshow("vis",vis)
cv.waitKey(0)
cv.destroyAllWindows()
```

结果如下：



可以看出这个居然是一一对应的，后面我输出了关键点的坐标和特征向量，发现居然全部都是是一样的。后面对于两张图，我使用了不同的模型去处理，得到的结果如下：



发现效果依旧很差，然后就不知道该怎么去处理了。

结果分析与体会：

这个实验没有完成，不是很了解 r2d2 模型的代码跑出来的那些数据为什么是不变的。