

《计算机视觉》实验报告

姓名：冯俊佳 学号：23122721

实验 8 图像检索

一. 任务 1

a) 核心代码:

```
01 filelist = get_imlist('/Users/feng/Desktop/计算机视觉/Week9/landmarks/data')
02
03 # 输入的图片
04 im1f = '488.jpg'
05 im1 = array(Image.open(im1f))
06 sift.process_image(im1f, 'out_sift_1.txt')
07 l1, d1 = sift.read_features_from_file('out_sift_1.txt')
08
09 i = 0
10 num = [0] * len(filelist) # 存放匹配值
11 for infile in filelist: # 对文件夹下的每张图片进行如下操作
12     im2 = array(Image.open(infile))
13     sift.process_image(infile, 'out_sift_2.txt')
14     l2, d2 = sift.read_features_from_file('out_sift_2.txt')
15     matches = sift.match_twosided(d1, d2)
16     num[i] = len(matches.nonzero()[0])
17     i = i + 1
18     print('{} matches'.format(num[i - 1])) # 输出匹配值
19
20 i = 1
21 figure()
22 while i < 4: # 循环三次，输出匹配最多的三张图片
23     index = num.index(max(num))
24     print(index, filelist[index])
25     lena = mpimg.imread(filelist[index]) # 读取当前匹配最大值的图片
26     # 此时 lena 就已经是一个 np.array 了，可以对它进行任意处理
27     # lena.shape # (512, 512, 3)
28     subplot(1, 3, i)
29     plt.imshow(lena) # 显示图片
30     plt.axis('off') # 不显示坐标轴
31     num[index] = 0 # 将当前最大值清零
```

```
32         i = i + 1
33     show()
```

词袋口优化:

a) 核心代码:

```
01 # 构建视觉词典
02 def build_visual_vocabulary(features, num_clusters):
03     kmeans = KMeans(n_clusters=num_clusters)
04     kmeans.fit(features)
05     visual_vocabulary = kmeans.cluster_centers_
06     return visual_vocabulary
07
08 # 计算图像的词典表示
09 def compute_bovw_representation(image_descriptors,
10     visual_vocabulary):
11     bovw_representation = np.zeros(len(visual_vocabulary))
12     nearest_neighbor = NearestNeighbors(n_neighbors=1)
13     nearest_neighbor.fit(visual_vocabulary)
14     distances, indices =
15     nearest_neighbor.kneighbors(image_descriptors)
16     for index in indices:
17         bovw_representation[index] += 1
18     return bovw_representation
19
20 # 计算两个词典表示之间的相似度
21 def compute_similarity(bovw1, bovw2):
    similarity = np.dot(bovw1, bovw2) / (np.linalg.norm(bovw1) *
    np.linalg.norm(bovw2))
    return similarity
```

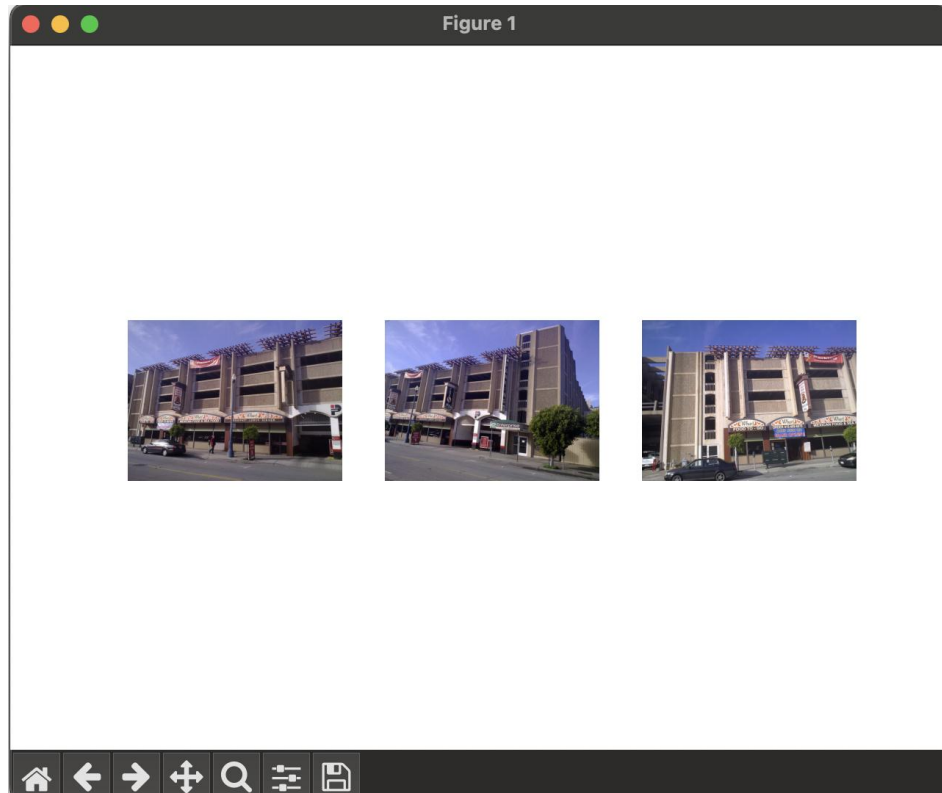
注:

测试图片集/训练的图片集: 地标建筑图片集, 约 500 张选取

<https://purl.stanford.edu/rb470rw0983> 网站的 lansmarks 中的 Query 为数据集。

b) 实验结果截图

检索结果按照相似度进行排序:



终端输出结果:

```
processed tmp.pgm to out_sift_2.txt
0 matches
processed tmp.pgm to out_sift_2.txt
3 matches
processed tmp.pgm to out_sift_2.txt
0 matches
processed tmp.pgm to out_sift_2.txt
2 matches
processed tmp.pgm to out_sift_2.txt
2 matches
processed tmp.pgm to out_sift_2.txt
0 matches
processed tmp.pgm to out_sift_2.txt
1 matches
```

c) 实验小结

经历了前几次的实验，基本上实验 8 图像检索也是同样一个道理。在计算机视觉的实验过程中，往往是先确认实验的一个背景目的，选取相应的一个模型，找合适的数据集，进行划分之后开始训练，最后得出结果。无论是之前的人脸识别还是人脸检测还是这个图像检索，需要在整个实验的过程中去感受解决问题的一个过程，并在不同的实验中逐渐熟悉各个库和算法的实际使用。

在这之中不仅收获了对于计算机视觉各个方面的应用的经验,还提升了自己对于信息的检索能力。