数字图像处理第二次作业

任务1 ：基于kmeans 的sift特征的加速优化

实验有三个部分组成，task1.m，siftmatch.m，kmeansAcc.m

有task1.m调用sitfmatch.m 进行两幅图像相似特征的优化，并设置标志位accFalg来决定是否使用kmeans加速优化。实验步骤如作业说明所介绍的，这里不再叙述。

我们来看看实验结果：

1基于传统的最小匹配：

耗时22.3s

2 基于kmeans的加速优化比配

好事11.2s

实验分析：

传统两幅图像A：m\*N1和B：m\*N2（sift特征维数，本实验是128）其时间复杂度为O(m\*N1\*N2)对于图像A每一个sift特征都与B中所有的sift的特征进行空间距离计算。利用kmeans进行加速，我们对128维的sift特征分成16分，对每一份的所有的特征进行kmeans聚类（k：64类，），其中kmeans的时间复杂度为O(Iter\*N1\*k\*d)iter是迭代 次数（matlab默认100），N是数据个数，k是聚类个数，d是每个数据的维度。我们需要对数据A和B分别聚类m/16次，本实验是8次。所以总共的时间复杂度为：(OA（kmeans）+ OB（kmeans）)\*8,计算16个中心distmatrix时间复杂的为16\*k\*k，相对较小。所以采用kmeans聚类 N很大的时候占有优势。粗略估计一下，本文中大概加速了一倍。

Task2

本实验有4个部分组成，分别是利用kmeans计算codebook的codebook.m和利用vlad方式将每幅图像合成128\*k的长向量的imvlad，还有图像匹配imsearch.m。实验步骤如说明所示，这里不再赘述。

实验结果:

实验分析

本实验一共20幅图像，利用kmeans对所有的图像进行聚类，这个耗时最多，复杂度为O(Iter\*sum\_N\*k\*d),其中iter是迭代次数（matlab默认100），N是所有图像sift特征和，k是聚类个数（本文16），d是每个数据的维度（本实验128），再分析一下imvlad对图像特征聚合，对于每一个我们需要计算N\*k次，进行聚合，相比kmeans相对较小。