**简单的图像检索算法**

**一、试验任务描述**

1）使用CIFAR-10的一个子集作为图像检索数据集（包括query和database）

2）**使用颜色特征（RGB直方图）和纹理特征（LBP）这两种特征，**采用余弦相似度作为图像的相似度度量，设计一种简单的图像检索算法

3）设计一个具有简单GUI的图像检索demo系统（包含图像调入、检索返回top-20图像列表等功能）

**二、技术方案**

**1.颜色特征**

**1）总体思路**

①读入一张指定的query图像，计算其RGB特征向量

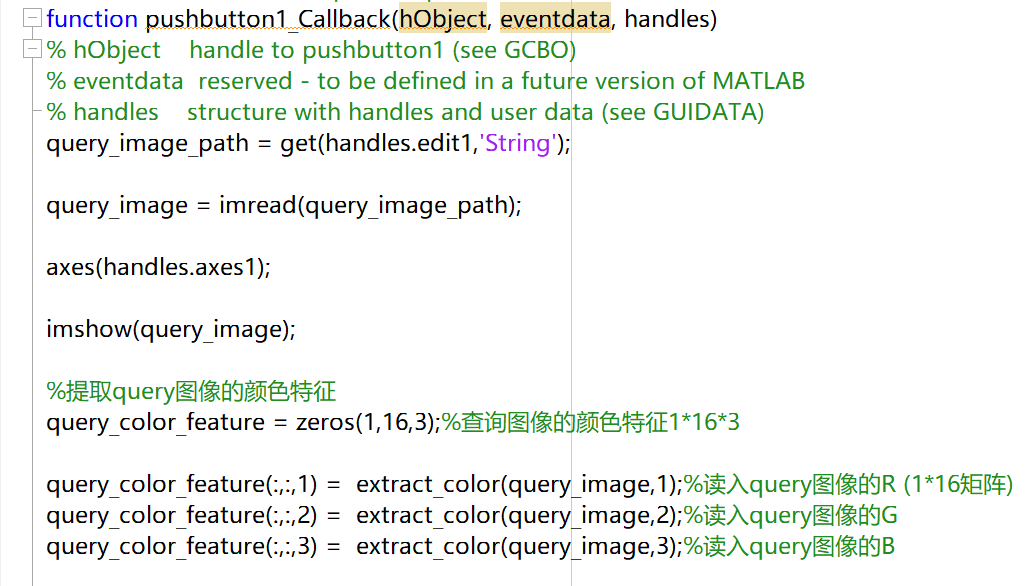
* 其中特征向量为产生的RGB直方图的纵坐标；直方图的横坐标则一共是16，即将256个灰度级平均分为16组，统计落入每一组的像素数量
* 一张图片的特征向量有3个，分别是R向量、G向量、B向量，每个向量是1\*16

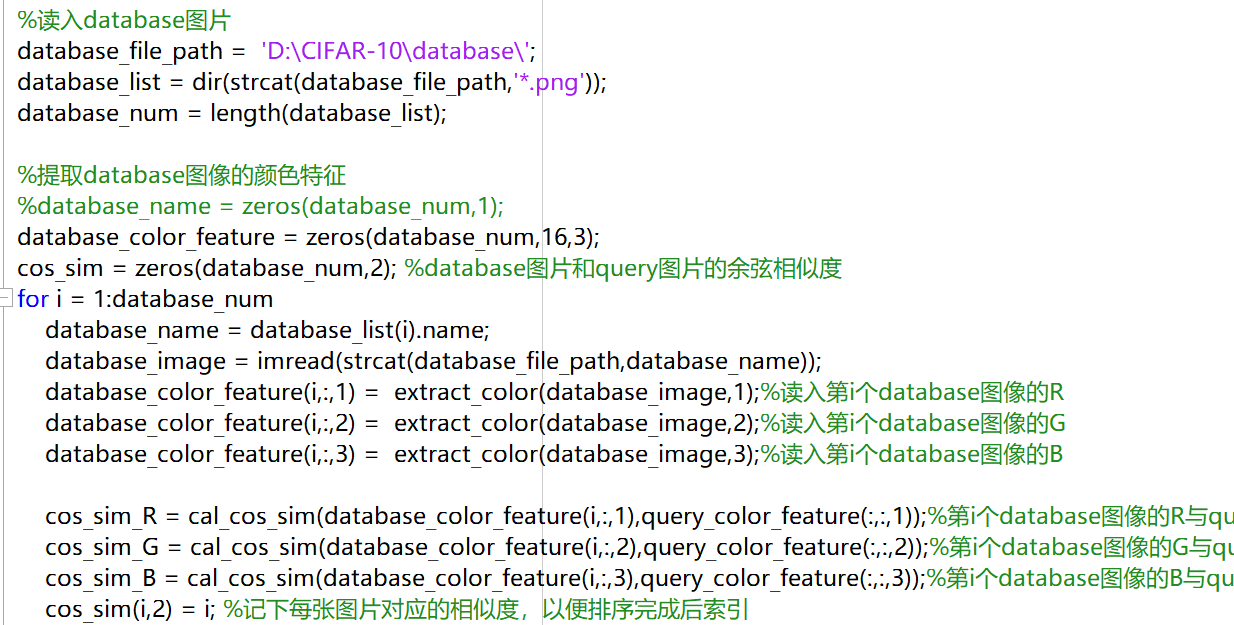
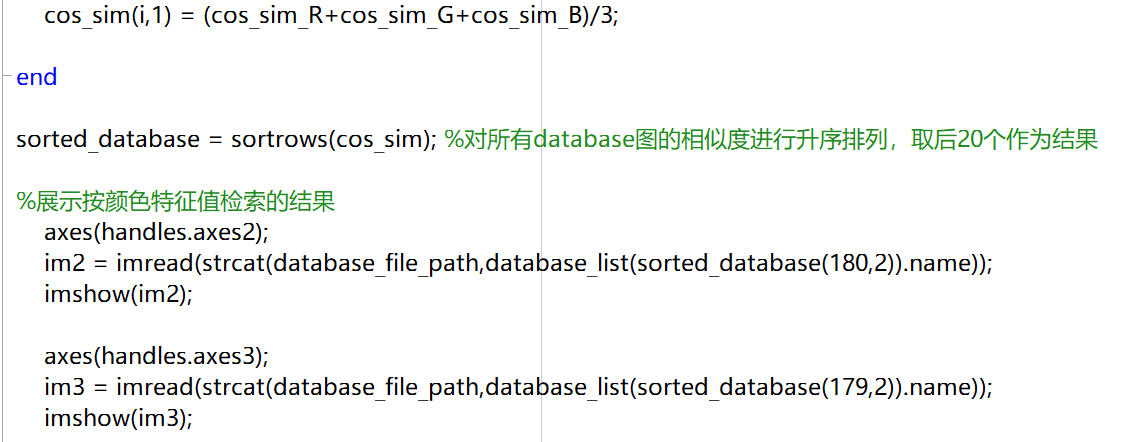
②依次读入总共180张database图片，分别计算其RGB特征向量，并全部存下来，显然特征矩阵是180\*16\*3的

③每计算出一张图片的特征向量，与query的向量做余弦相似度的计算

* database的R向量与query的R向量做余弦相似度的计算，G/B向量同理，因此会得到3个余弦相似值
* 将RGB三个维度的余弦相似值求平均，得出最终标志着该database图片与query图片相似度的唯一余弦相似值
* 将全部180个余弦相似值全部存下

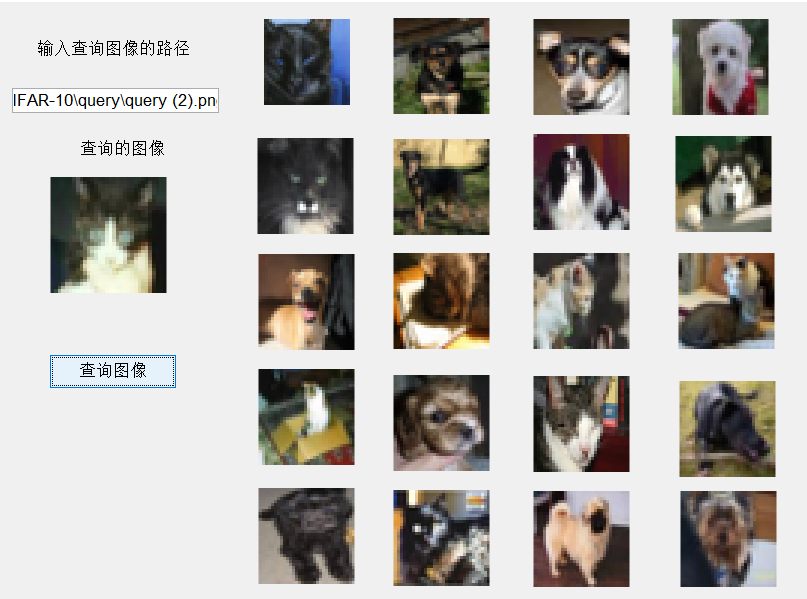
④对180个余弦相似值做升/降序排序，值越大，越相似，展示出最相似的20张图片

 **2）代码主程序**

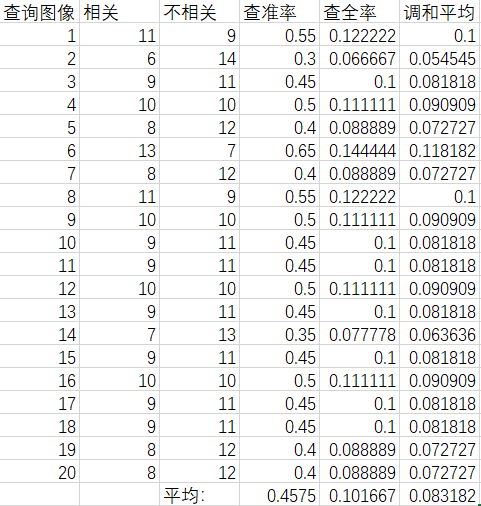


**展示代码省略了部分**

**3）结果展示**

 **示例查询query（1）：**

**4）实验结果分析**



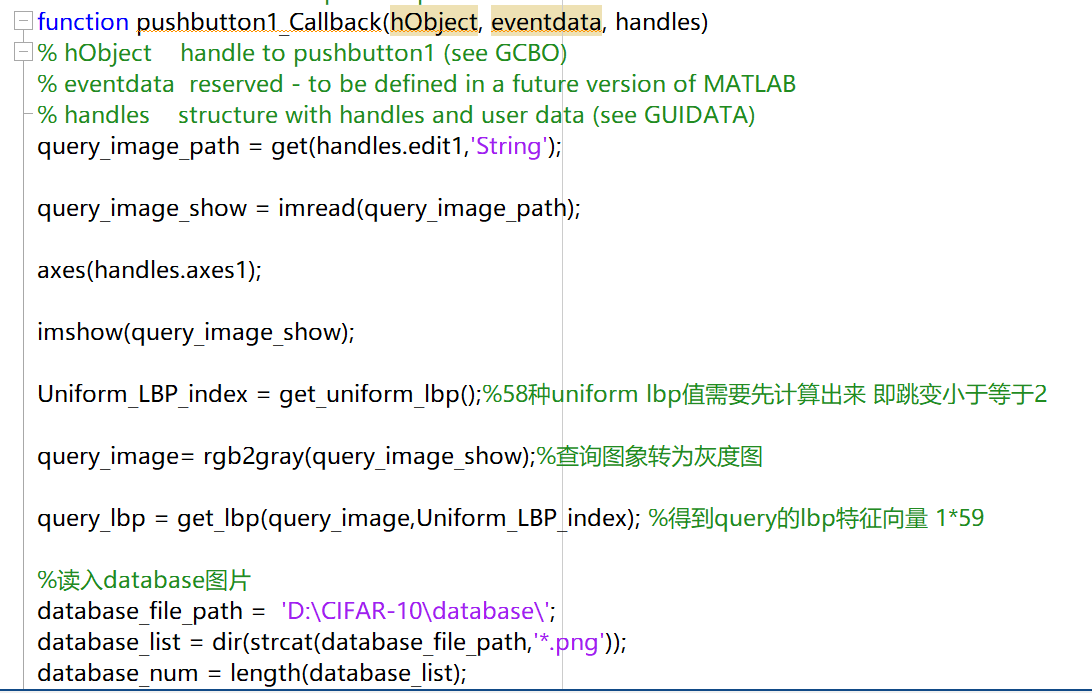
**2.LBP特征**

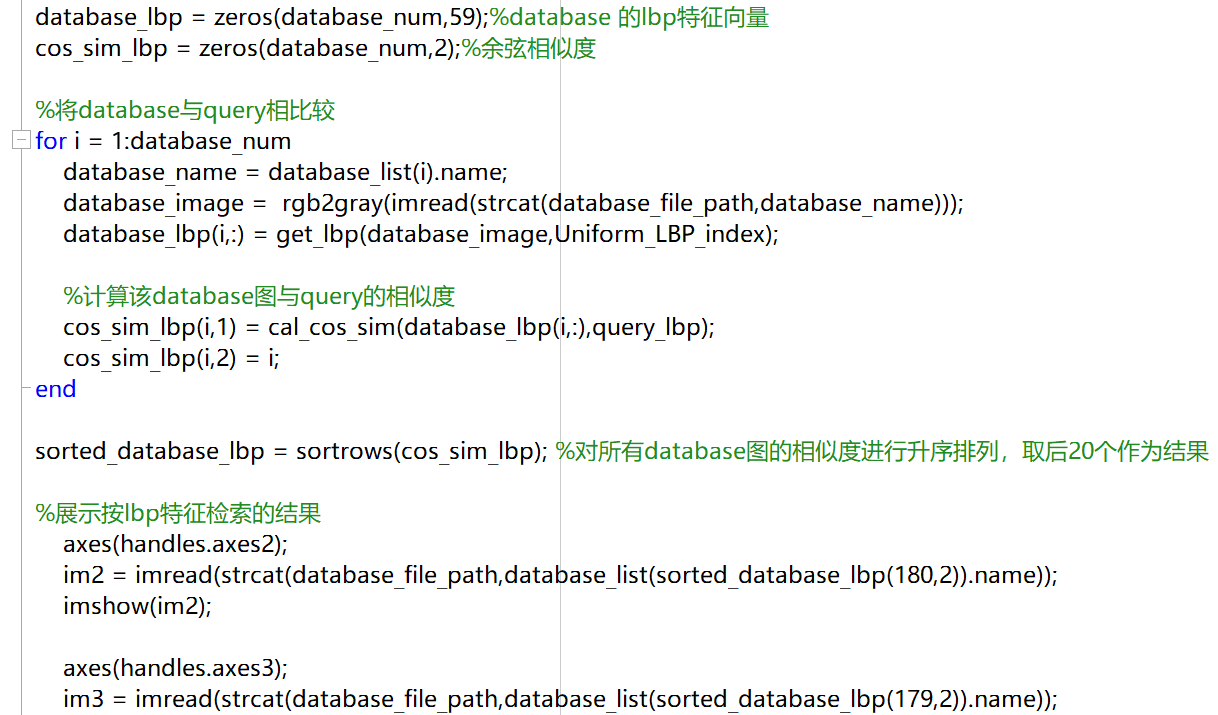
**1）总体思路**

整体过程与颜色特征几乎相同，不同点有：

①特征向量为LBP直方图的纵坐标，横坐标为59种LBP值情况，统计落在每种LBP值上的像素个数

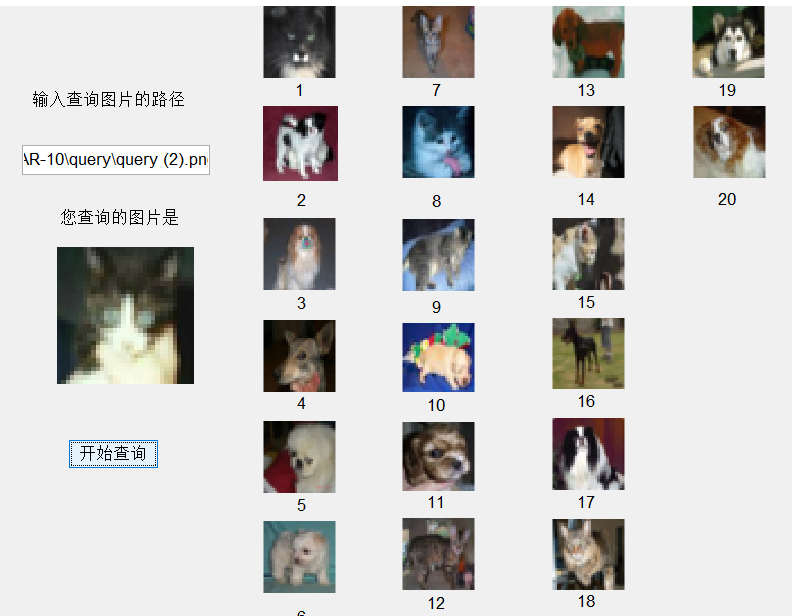
* 前58组LBP值为其二进制数跳变数小于等于2的所有数，第59组是二进制数跳变数大于2的所有值，这是uniform lbp的定义
* 每张图的LBP特征向量是1\*59

**2）代码主程序**

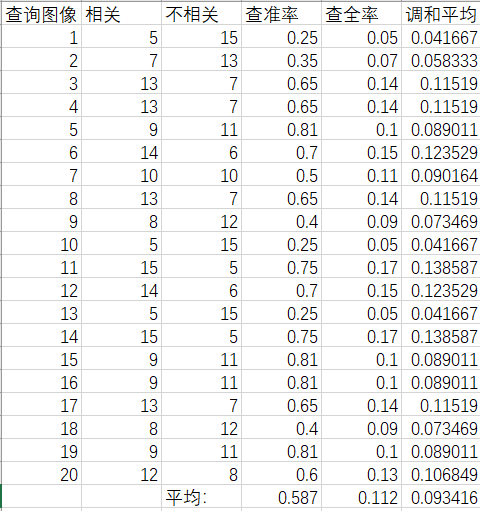


**展示代码省略了部分**

**3）结果展示**

 **示例查询query（2）：**

**4）实验结果分析**



**三、实验总结**

本次实验用颜色特征和LBP特征做了简单的图像检索算法和简单的GUI界面。

从实验结果上来看，准确率并不高，且颜色特征的准度比LBP略低，这是按照各种指标定义算出来的结果；但肉眼看上去，颜色特征查找的图片颜色确实与query非常相近，但lbp则不好判断。

实验存在的问题：lbp算法存在问题，每一张图的lbp值属于59组的数据都非常多，因此推断所设计的lbp算法存在一定的问题

gui的fig文件无法直接双击打开后正常工作，而是要在matlab的gui工作台打开运行，才能正常工作，存在一定的问题