课程实验大作业 1

学号：3180106177　　姓名：　程诗卓　　班级：求计 1801

实验要求：

任意给定一张书法字的图像，请编写程序自动判别图像并输出数据集中相同字的图案。

备注：

(1) 老师提供数据集；

(2) 每个同学独立编写代码实现；

(3) 按照识别准确率给分；

实验内容：

方法描述：

一、　实验原理

以下理论来自于《根据形状相似性的书法内容检索》（中图分类号 TP39）

1. 轮廓点的形状属性：

只判断轮廓的特征信息较判断所有像素信息计算量少，因此统计轮廓点。

对于轮廓上的像素点p，以其为中心做极坐标系，以r＝4、r＝8、r＝16 、r＝32和每个区间pi/4的角度，将周围区域分为32个区域，类似雷达，如图。

通过统计各区域内的轮廓点数，获得一个32*1的矩阵，这就是该像素的形状属性。区域越靠外r越大，因为越远的点，对于两图中匹配点选取的其重要性越低。

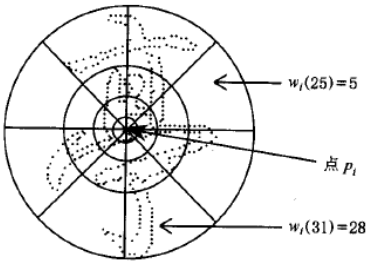

图 2　描述书法字像素点属性的坐标系

2. 轮廓点的匹配度：

对于样本字m中的某一点m_i，其对待匹配字n中的某一个点n_j的匹配度C_{ij}由下式确定，其中w为形状属性中的分量。

如果满足前提条件：（某点的匹配点有一定绝对范围，不会出现从一个角落到另一个角落的情况）

$$dis = |m_i - n_j| = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \leqslant \sigma \times length$$

则可以计算两点的相似度：

$$C_{ij} = C(m_i, n_j) = \frac{1}{2}\sum_{k=1}^{32}\{[w_i(k) - w_j(k)]^2\}/\{w_i(k) + w_j(k)\}.$$

因为匹配度C一定存在一个最小值，其对应的n中的像素就是m所对应的最佳匹配点。

$$PMC_i = \min\{C(m_i, n_j): j = 0,1,2,\cdots,n\}.$$

在两字完全相同的情况下，每一点的最小匹配度PMC都为0；

3. 单字的匹配度

计算样本字中各个点对相应点的匹配度之和加上与对应点的欧氏距离之和（乘上a，经验系数），匹配度越高该值越低，通过对该值排序可得形状相似度的排序。

$$C_{total} = \sum_{i=1}^{n}(C_i + a\|p_i - p_i'\|^2)$$

二、预处理

Batch 批处理函数

通过 Uigetdir 用户窗口获取数据集的地址（点击“数据集”作为根目录）。并通过字符串连接为*.gif 为结尾，以此，可用 dir 读取数据集中所有（可选择 kai 楷书 cao 草书界定二级目录从而只读取其中一种字体）gif 的文件信息存为 im_path_list.，使用该结构中的 name 与 folder 获取图片具体地址，imread 读入图像，存在元胞数组 set 中。

输出：

1.将图片通过 im2bw 转为二值图像（graythresh 自动获取灰度阈值）存在 set 的第 1 行。

2.将文件名存在 set 的第 6 行。

data_read 函数：读取文件以及获取轮廓特征属性并存储。

（1） 将 set 中图像缩放为 32*32 的二值图像，去除一些不必要信息以及减少运算量。（但可能会造成重要细节丢失，降低查准率）存储为第 1 行

（2） catch_shape 函数通过判断像素与周围像素的差值（上下左右只要有一个方向差为 1 即为轮廓点），获取轮廓信息，存在第 2 行。

（3） character 函数计算轮廓的属性值，存在第 4 行。

（4） 第 3 行存储该字的属性矩阵长度。

最终数据格式：

1.压缩图（二值化）

2.轮廓坐标

3.轮廓点数

4.轮廓点形状属性

5.原图（二值化）

6.文件名

	1
1	32x32 logi...
2	139x2 dou...
3	139
4	139x32 do...
5	370x370 lo...
6	'何绍基_02....

存储为.mat 文件，之后检索只需读取之。

三、样本读取、检索。

run_test 测试函数：

载入预处理后的数据集，ui 界面获取 test 样本文件，按照预处理的方法处理之。

用 compare_c 依次计算 test 对于数据集中每一个字的匹配度（公式见前），返回匹配度与其坐标。

对各点匹配度进行升序排序，返回[B L]排序结果以及标签。

用户输入参数 num threshold trial

由于对于不同样本字，其对数据集的匹配度序列不一样，且难以找到一个算法得到满意的匹配度的阈值，使得展示的字中查准率和查全率的组合比较理想，所以本次实验采用用户输入所需最大展示字数、搜索区间、尝试次数来确定展示的字。

Num：最大展示字数，暂时设定窗口是 6*6 个图形的空间，因此 num 不能大于 35（第一个位置展示样本）

Threshold：由于相似度算法一定能够找到匹配度最高的一个字，若样本在数据集中选取，则该算法能保证一定就是原图。而数据集的读入具有顺序，所以在原图前后寻找匹配字则可以找到与原图相同的字。（草书和楷书交错存储）（如果不需要则可将 threshold 设为一个大数）一般设定为数据集中该字的数目

Trial：尝试次数，代表了在排序后的匹配值序列中的前 trial 个字中，寻找满足 threshold 设定区间的字。一般要设定相对较大。

实验结果与分析：

通过设定检索区间，以匹配度最高的为基准在周围进行检索，保证了差准率，而查全率可以通过提高尝试次数来提高：

若不设定检索区间，匹配度排序确定的序列中，前 n 个字有 m 个是正确字，应该匹配到的总字数是 a，则查全率为 m/a，查准率为 m/n。

设定检索区间，将范围缩小到范围为 2*threshold 的区间，跳过了序列前部存储位置差距巨大的字，而存储时草书文件夹和楷书文件夹互相交错，很大程度上减少了错字的发现。因此要提高查全率只需提高 trial，即在序列的前 trial 个字中搜索。

此方法的弊端是：若匹配度最高的字为错字，则之后所查全为错字。

本次检索，目的是在楷书中检索楷书，在草书中检索草书，但由于形状匹配也会造成草书和楷书的匹配，所以会出现混杂。若需要混合检索，只需调高 threshold 即可。

总之，本文提供了一个用户自定义参数来检索相似书法字的 MATLAB 程序，用户使用上包括 ui 界面批处理数据，ui 界面读取测试图，输出匹配图案以及名称等功能。

内部实现了图像归一化处理、形状属性提取、图像匹配的功能。

理论来源为：《根据形状相似性的书法内容检索》**（中图分类号 TP39）《计算机辅助设计与图形学学报》第 17 卷 第 11 期**

查全率-查准率曲线由于统计工作需要数据量，而检测正确率需要人工，时间紧迫，无法绘制。以下是一些运行实例。

简体无：查全率100%,查准率50%（调低显示个数即到100%因为前4个都是）

重：查全率100%,查准率100%
该字特征比较明显

月：查全率100%,查准率50%（调低显示个数可至72%）

仰：查全率83%,查准率100%

待加入的权重：提取骨架（粗细笔画、连笔识别）
（这次识别有可能是提取图片时压缩，细笔画像素较少导致缺失）

繁体节：查全率87.5%(7/8不算简体)
查准率87.5%

繁体简体差别过大无法识别

本次实验采用了设定检索区间的方法保证了检索的正确率，但前提是数据集的字符存储是有序的，如果仅依靠形状相似度，以下因素会降低查准率：

1） 图像压缩损失细节（细笔画、密集笔画等）；
2） 区域判断函数 my_area 中的雷达区间划分长度为经验划分；
3） 单字匹配度的欧氏距离的权重 a 为经验值；
4） 匹配点判断时，进行判断前提条件为距离小于某个值（isClose 函数），该值也为经验值；
5） 汉字的形似字不易区分：日和月、丈和万、散和众（繁体）等；
6） 某个字中含有样本字的一部分（子图也可能判断相似），如有和月。

去除检索区间，仅有形状相似度的检索：

者：查全率50%(7/14),查准率37%(7/19)
（形状相似的月字以及带月字的有字）
（草书形状过于诡异）
（枝与节字在压缩后的轮廓细节损失较多）

总之大体结构是上下空格，中间连接的矩形（日字轮廓）

众：查全率50%(7/14),查准率50%(7/14)
（形状相似的散字以及春字）
（压缩后的轮廓细节损失较多）

大体结构是上部小空格，中下部收拢，下部展开（撇与捺）

此外，图像二值化处理时，由于 grayshresh 的不完善，个别图像会与其他图反相。但不影响轮廓提取与后续计算。