

数字图像处理综合作业 2 报告

一.细化前操作：二值化，闭运算，去孤岛

二值化阈值 0.498，是使 r96_4.bmp 外围变白的最大值，这样两张图处理后都是外围是白色的，原图的黑色为指纹。



白色的闭运算是黑色的开运算，然后通过连通域大小来去除黑色中白孤岛和白色中黑孤岛。



二.细化

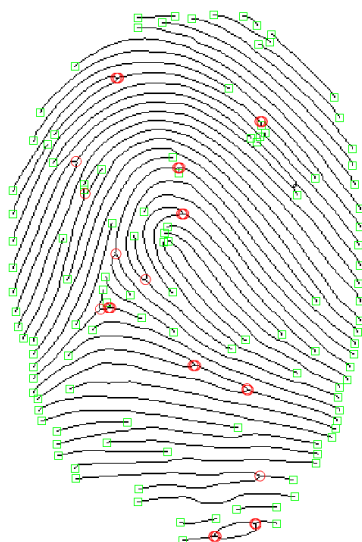
直接用~bwmorph(~lc,'thin',5)函数



三.细化后处理

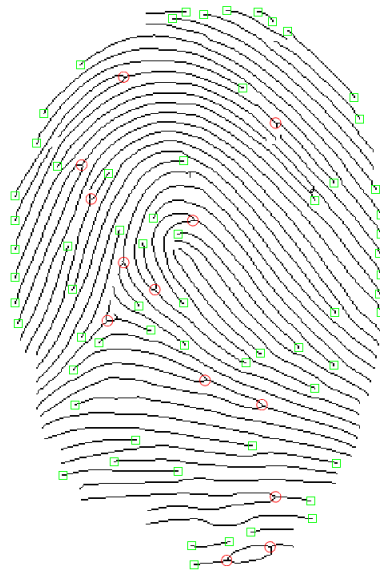
由于最初选择了对黑色而言的开运算，所以黑色的已经连得较松，有少量甚至断开了一点，我尝试过 bwmorph 里 spur 的剪枝，也自己写过一个剪 H 的函数，都存在着一个问题——毛刺和桥接剪去的同时，一些分岔点也被剪了部分像素，就断开了 1, 2 个像素这样的，导致后续识别中无法将其识别为分岔点。那么就要写函数将其补上，我也试过通过特征图样的匹配和在像素点周围走一圈看与黑边交几次的方法，都存在图上有其它较多位置也符合特征从而被错误填充了黑像素的问题。且分岔点的方向不好获取或判定，着实不好补充。所以处理的原则就是：不能剪多，剪多了失去的特征就恢复不回来了，要保证所有特征点都被识别出来；剪少可以，自己通过算法来判定错误识别的特征点，将他们去除！

特征点识别就是看黑像素周围 8 个像素里黑像素的个数，1 就端点，3 就分岔点，写了函数造一个 lut，然后 bwlookup 一下。



首先全部识别出来了，可以看到分岔点容易在一个位置识别多个，先把重叠的清除了。再有，短线毛刺不在细化图除，会被同时识别为端点和分岔点，可通过这一特征将其去除。再有，有的端点可能是本来一条线的，最初没连扎实，闭运算时给它弄断了，也可以通过他们间的距离把他们去除。

这个顺序就很讲究。挨在一起的交叉点先去，因为它也可以是毛刺导致的重合，不然去毛刺后会留下一个；去毛刺导致的端点和交叉点挨一起时，也要注意不能分岔点和端点挨一起就去，还要检查是否端点旁还有端点，因为可能是分叉点没连上断开了一点而不是毛刺！最后去挨在一起的端点还不能去挨在一起的分岔点的，因为前者全去掉，后者要留一个。



四.去边界点

So easy! 直接对每个端点，往上下左右四个方向走，有一个方向走完没遇到一个黑像素，就是边界点，去掉就 ok 了

