实验报告

14301027 汪棋

T1

算法描述：

图像扩张：

含义：a和b叠加后至少有一个点为1（叠加运算在代码里为与运算）

图像腐蚀：

含义：c和b叠加后点都是1（叠加运算在代码里为与运算）

在给出模板及对应中心点后，用上述算法（类似卷积操作）对原图像的每一个像素进行操作，从而得出扩张或腐蚀的图像

代码见文件（prob1.m dilation.m erosion.m）

结果如下：

（a）

0 0 0 0 0 0 0

1 1 1 1 0 0 0

0 1 1 1 0 0 0

0 1 1 1 1 0 0

1 1 1 1 1 1 0

1 1 1 1 1 0 0

1 1 1 1 1 1 0

0 0 0 0 0 0 0

（b）

0 0 0 0 0 0 0

0 0 0 0 0 0 0

0 0 0 0 0 0 0

0 0 0 0 0 0 0

0 0 1 1 0 0 0

0 0 1 0 0 0 0

0 0 0 0 0 0 0

0 0 0 0 0 0 0

（c）

0 0 1 1 0 0 0

0 0 1 1 1 0 0

0 0 0 1 1 0 0

0 0 1 1 1 1 0

0 0 1 1 1 1 1

0 1 1 1 1 1 0

0 1 1 1 1 1 1

0 0 0 0 0 0 0

（d）

0 0 0 0 0 0 0

0 0 0 1 0 0 0

0 0 0 0 0 0 0

0 0 0 0 1 0 0

0 0 0 1 1 0 0

0 0 0 1 0 0 0

0 0 0 0 0 0 0

0 0 0 0 0 0 0

（e）

用第一个模板

0 0 0 0 0 0 0

0 0 0 0 0 0 0

0 0 0 0 0 0 0

0 0 0 0 0 0 0

1 1 1 1 0 0 0

1 1 1 0 0 0 0

0 0 0 0 0 0 0

0 0 0 0 0 0 0

用第二个模板

0 0 0 1 0 0 0

0 0 0 1 1 0 0

0 0 0 0 1 0 0

0 0 0 1 1 1 0

0 0 0 1 1 1 0

0 0 0 1 1 0 0

0 0 0 0 0 0 0

0 0 0 0 0 0 0

（f）

用第一个模板

0 0 0 0 0 0 0

1 1 0 0 0 0 0

0 1 0 0 0 0 0

0 1 1 0 0 0 0

1 1 1 1 0 0 0

1 1 1 0 0 0 0

1 1 1 1 0 0 0

0 0 0 0 0 0 0

用第二个模板

0 0 0 1 0 0 0

0 0 0 1 1 0 0

0 0 0 0 1 0 0

0 0 0 1 1 1 0

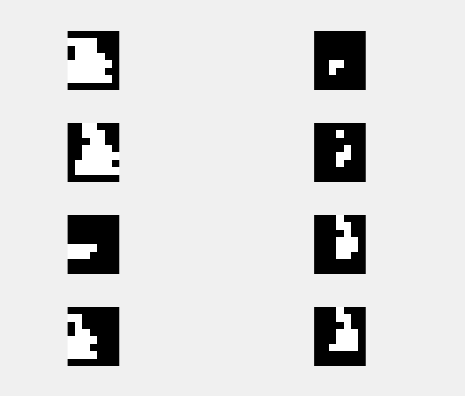
0 0 0 1 1 1 0

0 0 1 1 1 1 0

0 0 0 0 0 0 0

0 0 0 0 0 0 0

效果图如下（行优先）（上面的矩阵，可能复制错，以下图为准，图是不会错的）



T2

算法描述：

1. 利用全局阈值处理对光照均匀的图片进行图像分割：
2. 对阈值T取初值为图像像素平局值
3. 利用T进行分割，将图像分为G1和G2两部分
4. 取新的T为上述两部分的灰度值平均值的和的一半（T=（m1+m2）/2）
5. 若更新后的T与更新前的差小于某个值（如0.01），转到5，否则回到2
6. 用最终得到的T进行图像分割并输出结果（为二值图像）

2、利用动态阈值处理对光照不均匀的图片进行图像分割：

对于光照不均匀的图片，可采用分块的方法和动态阈值的方法，我采用了动态阈值的方法。动态阈值即阈值T与像素位置xy有关，我采用的是对每一个像素取其一个一定大小的邻域（大小为51时较效果好），计算其平局值为该点的阈值，并进行图像分割。此法不进行迭代

代码见prob2.m

效果图如下（左右分别为1和2）

