



学院：数据科学与计算机学院 年级专业：17 级计算类 科目：数字图像处理
学号：17341213 姓名：郑康泽

形态学和图像分割

1. A binary array that represents a portion of a black and white image is given below. Perform the operations listed below on this piece pf image. Assume that all of the pixels that surround this segment contain black background.

0	0	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	1	0	1	0	1	0
0	0	0	0	0	0	0

- (a) Dilation with the structuring element

①	1	1
---	---	---

 . The origin is shown with a circled element.

- (b) Erosion with the structuring element

①	1	1
---	---	---

 .

- (c) Dilation with the structuring element

1	①
0	1

 .

- (d) Erosion with the structuring element

1	①
0	1

 .

- (e) Opening with each of the above structuring elements.

- (f) Closing with each of the above structuring elements.

- (a) 算法描述：

- 1) 先将结构单元 180° 翻转，这样原点就在(1, 3)；
- 2) 为了使结构单元的原点能压在上述矩阵的每一个位置上，对矩阵的进行填充，即末尾添加两列 0；
- 3) 将结构单元的原点压到填充后的矩阵的每一个可压的位置上，即不超出矩阵的范围，取出矩阵被压到的片段，然后判断结构单元上为 1 的位置对应到片段相同位置上是不是 1，如果有一个重叠位置上同时为 1，那么结构单元的原点压住矩阵的对应位置上置 1；（结构单元压住的矩阵是未修改值的矩阵，中间修改值应作用到新声明的大小与原矩阵相同的全 0 矩阵上）；
- 4) 做完上述步骤后得到的新矩阵即为膨胀后的矩阵；

举例说明：

当上述结构单元的原点压在矩阵的(2, 1)上，压住的片段为

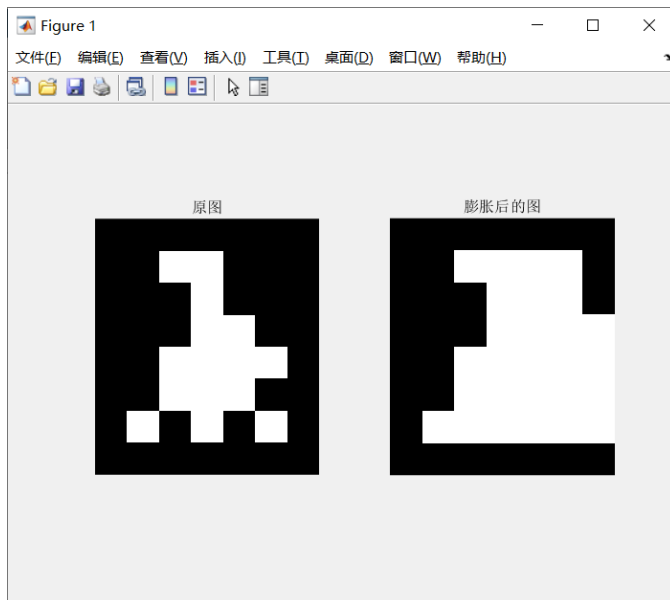
0	0	1
---	---	---

，因为结构单



元的第三个数为 1，所以满足条件，所以膨胀后的矩阵的(2, 1)的值为 1；

结果展示：



(b) 算法描述：

- 1) 为了使结构单元的原点能压在上述矩阵的每一个位置上，对矩阵的进行填充，即末尾添加两列 0；
- 2) 将结构单元的原点压到填充后的矩阵的每一个可压的位置上，即不超出矩阵的范围，取出矩阵被压到的片段，然后判断结构单元上为 1 的位置对应到片段相同位置上是不是 1，如果结构单元所有为 1 的位置对应片段的位置上也都为 1，那么结构单元的原点压住矩阵的对应位置上置 1 (结构单元压住的矩阵是未修改值的矩阵，中间修改值应作用到新声明的全 0 矩阵上)；
- 3) 做完上述步骤后得到的新矩阵即为腐蚀后的矩阵；

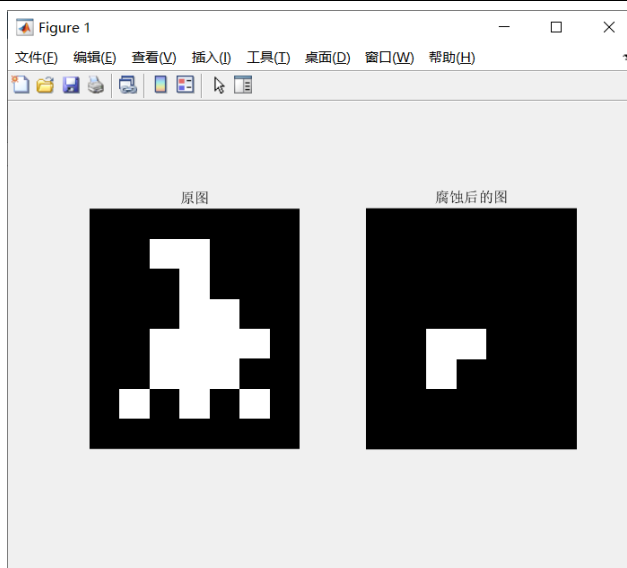
举例说明：

当上述结构单元的原点压在矩阵的(2, 3)上，压住的片段为

1	1	1
---	---	---

，因为结构单元和压住的片段全为 1，所以满足条件，所以腐蚀后的矩阵的(2, 3)的值为 1；

结果展示：



(c) 算法描述：

- 1) 先将结构单元 180° 翻转，这样原点就在(2, 1)；
- 2) 为了使结构单元的原点能压在上述矩阵的每一个位置上，对矩阵的进行填充，即最后添加一行 0 以及最后添加一列 0；
- 3) 将结构单元的原点压到填充后的矩阵的每一个可压的位置上，即不超出矩阵的范围，取出矩阵被压到的片段，然后判断结构单元上为 1 的位置对应到片段相同位置上是不是 1，如果有一个重叠位置上同时为 1，那么结构单元的原点压住矩阵的对应位置上置 1（结构单元压住的矩阵是未修改值的矩阵，中间修改值应作用到新声明的全 0 矩阵上）；
- 4) 做完上述步骤后得到的新矩阵即为膨胀后的矩阵；

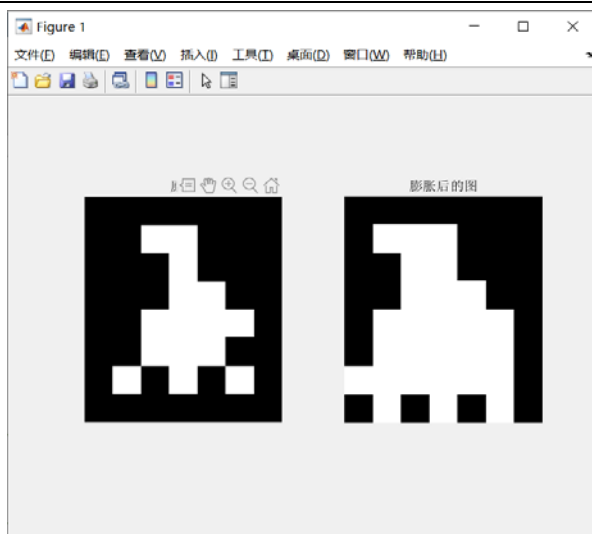
举例说明：

当上述结构单元的原点压在矩阵的(5, 3)上，压住的片段为

0	1
0	0

，因为结构单元的原点为 1，且压住片段的对应位置上也为 1，所以膨胀后的矩阵的(5,3)的值为 1；

结果展示：



(d) 算法描述:

- 1) 为了使结构单元的原点能压在上述矩阵的每一个位置上，对矩阵的进行填充，即最后添加一行 0 以及最后添加一列 0；
- 2) 将结构单元的原点压到填充后的矩阵的每一个可压的位置上，即不超出矩阵的范围，取出矩阵被压到的片段，然后判断结构单元上为 1 的位置对应到片段相同位置上是不是 1，如果结构单元所有为 1 的位置对应片段的位置上也都为 1，那么结构单元的原点压住矩阵的对应位置上置 1（结构单元压住的矩阵是未修改值的矩阵，中间修改值应作用到新声明的全 0 矩阵上）；
- 3) 做完上述步骤后得到的新矩阵即为腐蚀后的矩阵；

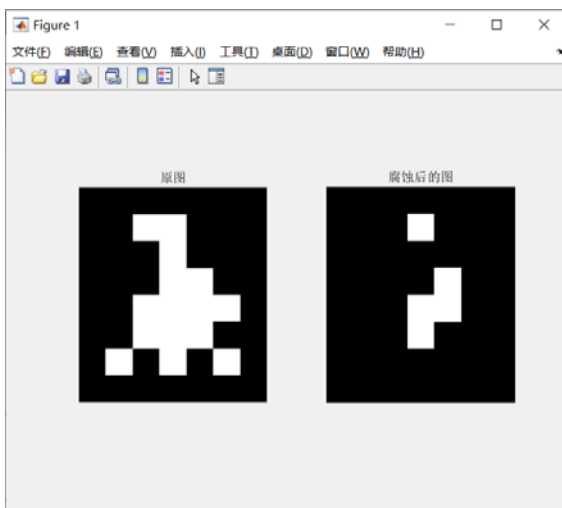
举例说明:

当上述结构单元的原点压在矩阵的(2, 3)上，压住的片段为

1	1
0	1

，因为结构单元为 1 的位置，压住的片段相应的位置为 1，所以满足条件，所以腐蚀后的矩阵的(2, 3)的值为 1；

结果展示:

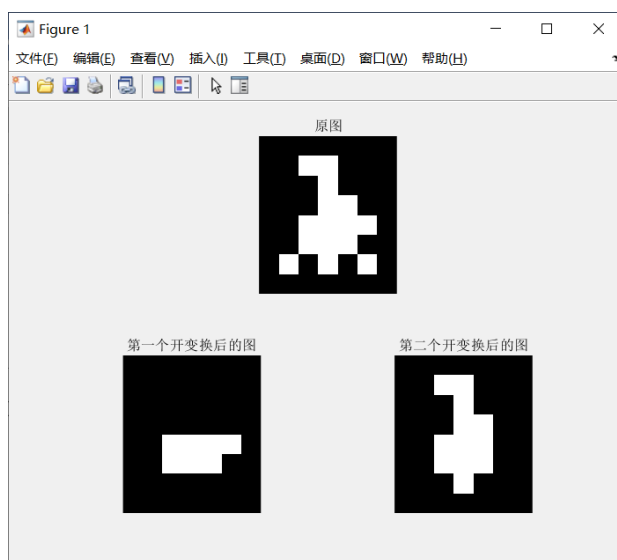




(e) 算法描述:

将原矩阵经过(b)和(a)即可得到利用第一个结构单元做开变换后的结果; 将原矩阵经过(d)和(c)即可得到利用第二个结构单元做开变换后的结果;

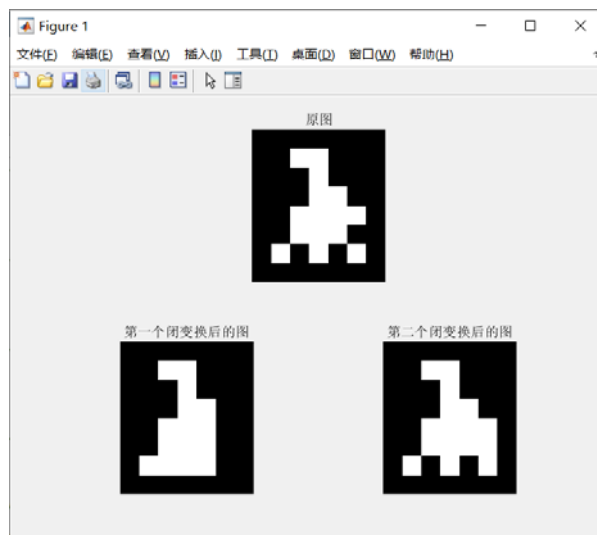
结果展示:



(f) 算法描述:

将原矩阵经过(a)和(b)即可得到利用第一个结构单元做闭变换后的结果; 将原矩阵经过(c)和(d)即可得到利用第二个结构单元做闭变换后的结果;

结果展示:



2. The images blobz1.png and blobz2.png are shown in figure 1. The difference is that blobz1 has nearly uniform illumination while blobz2 has very nonuniform illumination. The goal of this problem is to construct an algorithm based on



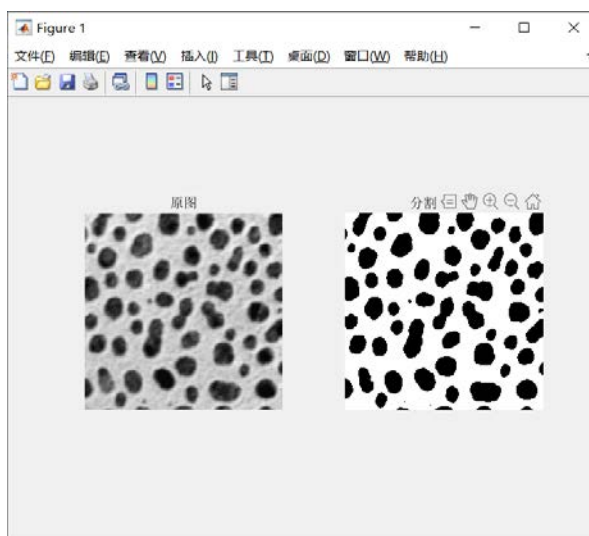
global greyscale thresholding for the segmentation of each image.

算法描述：

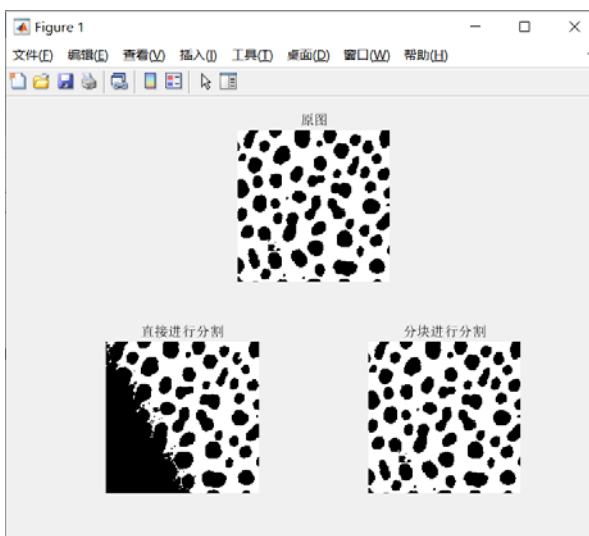
- 1) 选择一个阈值 T 的初始估计值（我选择初始估计值为原图的平均灰度值）；
- 2) 用 T 分割图像，生成两组像素：G1 有所有灰度值大于 T 的像素组成，而 G2 由所有灰度值小于或等于 T 的像素组成；
- 3) 计算区域 G1 和 G2 的所有像素计算平均灰度值 μ_1 和 μ_2 ；
- 4) 计算新的阈值 $T = \frac{1}{2}(\mu_1 + \mu_2)$ ；
- 5) 重复步骤 2) 到 4) ，直到逐次迭代所得的 T 值之差小于事先定义参数 ϵ ；

结果展示：

- 1) 均匀光照：



- 2) 不均匀光照：



因为直接进行基于全局的阈值分割效果不好，所以采取了分块后再进行全局的阈值分割，显然效果好多了。