《数字图像处理》
第11讲 图像分割(2)

冯建江

清华大学 自动化系

2017.12.14

内 容

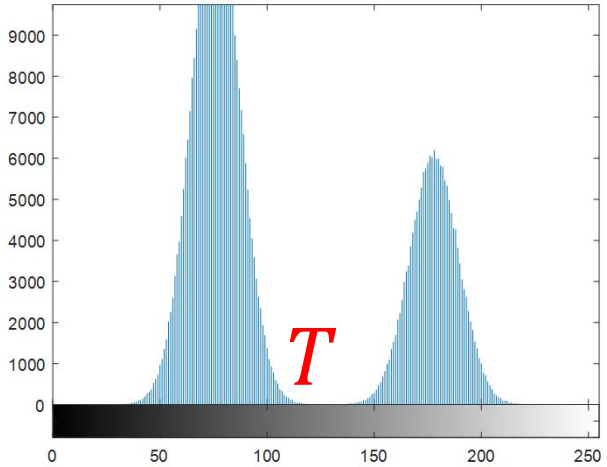
- 阈值法
- 基于区域的方法
- 分水岭法
- 其他方法

内 容

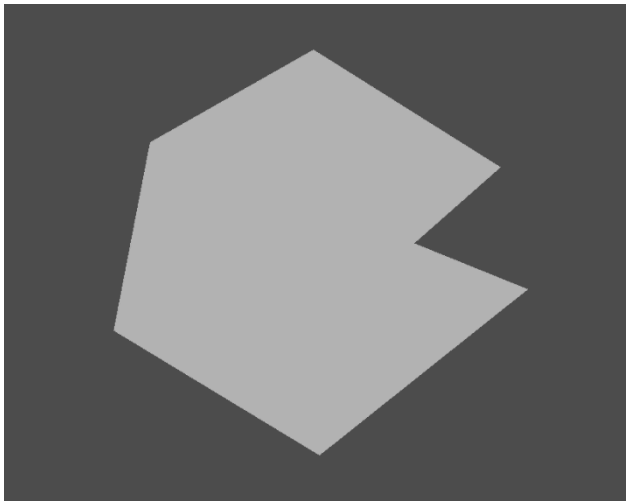
- 阈值法
- 基于区域的方法
- 分水岭法
- 其他方法

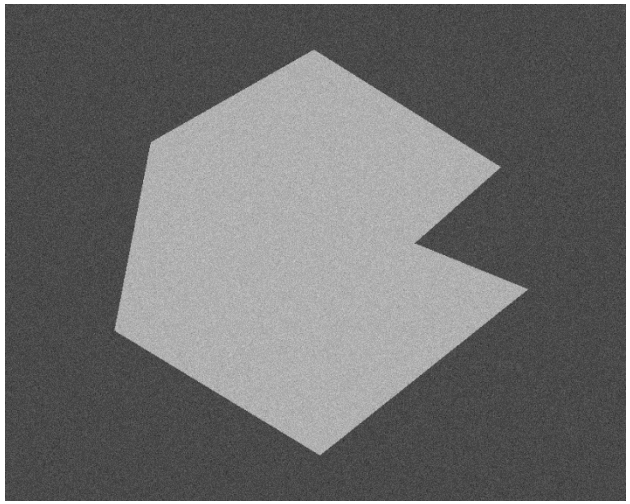
阈值法

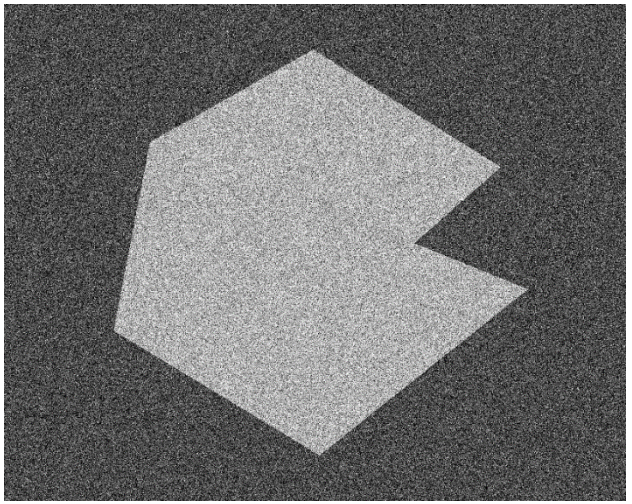
- 之前介绍的边缘检测方法，需将边缘连接成封闭轮廓，才能得到区域分割结果
- 阈值法直接将图像分为多个区域
- 假设图像的物体和背景灰度可分（灰度直方图如右图），使用一个全局阈值，就可得到满意的分割结果，叫全局阈值法
- 有时，阈值需要随局部特性变化，叫局部/可变/自适应阈值法
- 全局阈值法是否成功，受噪声、光照、物体反射特性的影响

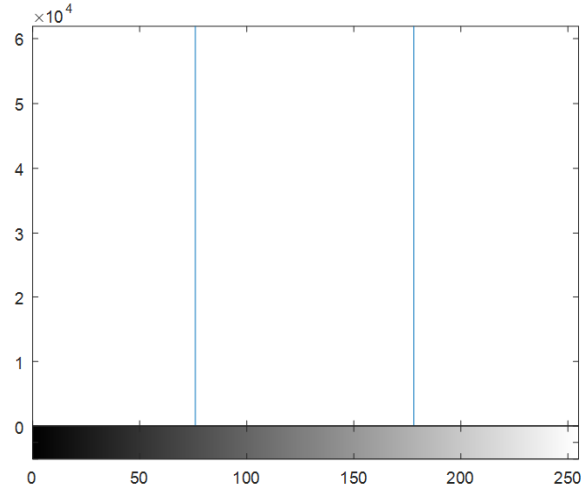


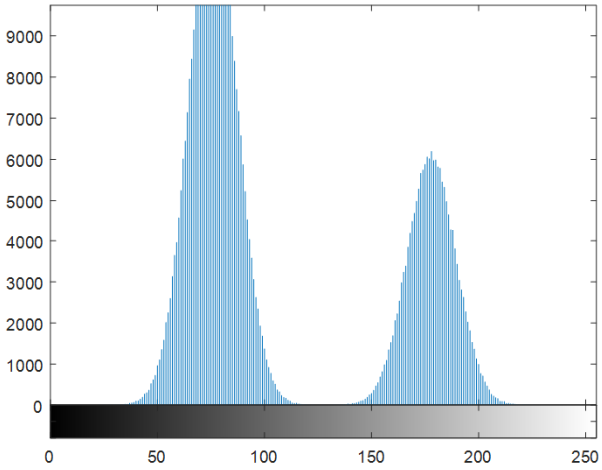
噪声对图像直方图的影响

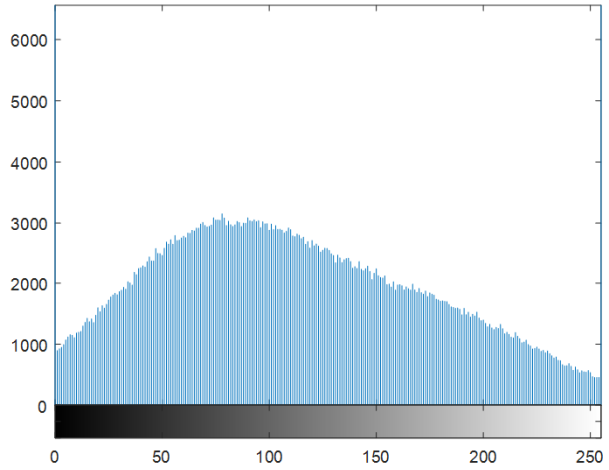




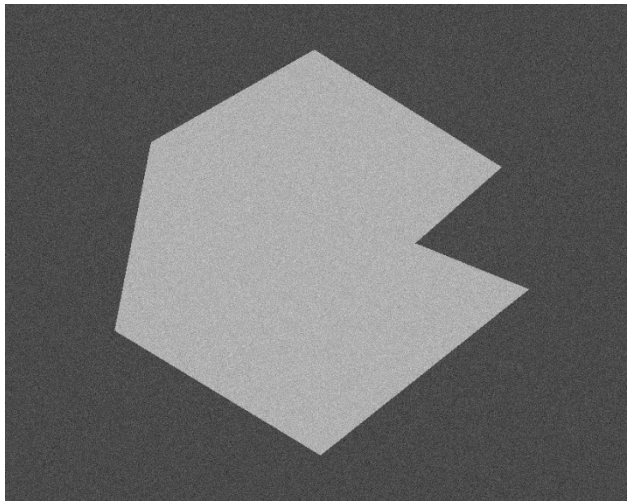


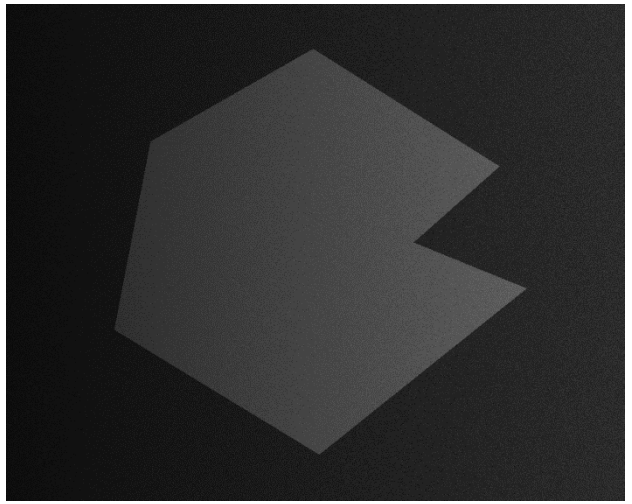


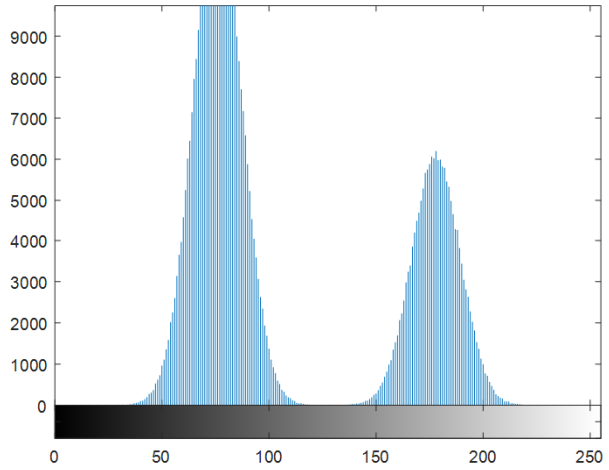


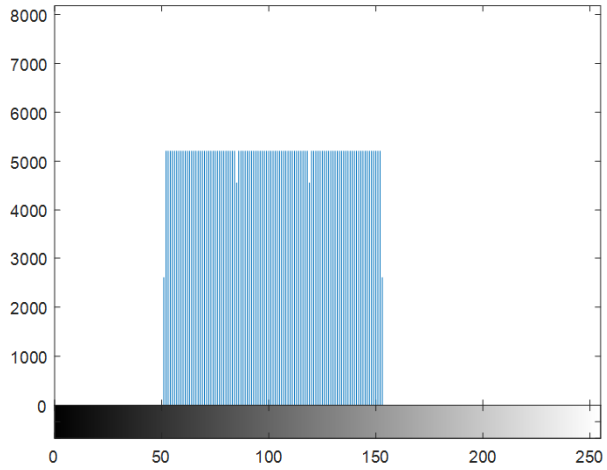


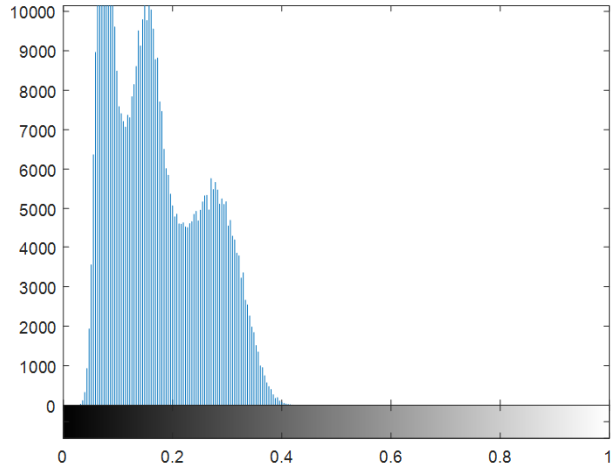
光照对图像直方图的影响









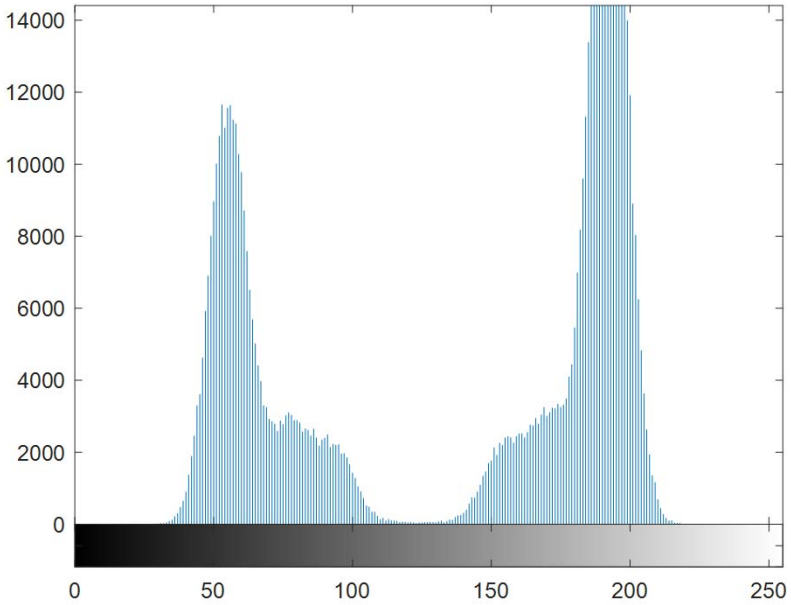


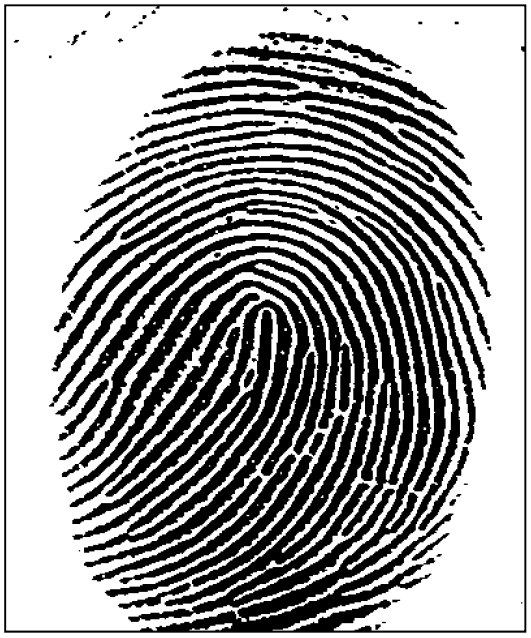
基本的全局阈值法

- 先考虑图像中物体和背景灰度可分的情况
- 由于不同图像的灰度直方图可能不同，所需的阈值因图而异，需要设计一个算法，自动选择合适的阈值
- 一种基本的全局阈值法：

1. 选择初始的T（例如图像均值）
2. 利用T对图像做二值化
3. 计算均值
4. 计算新的阈值T
5. 重复步骤2-4，直到T的变化小于某参数ΔT

全局阈值法的例子

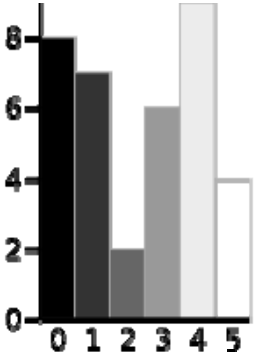


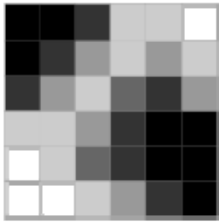


`ThresholdingIterative.m`

Otsu阈值法

- 思想：使类间方差最大的阈值
- 归一化直方图：$p_i = n_i/MN$
- 假设选择k为阈值，$0 \le k < L$
- 令C_1表示灰度≤ k的像素集合，C_2灰度> k的像素集合
- 类别C_1的概率为$P_1(k) = \sum_{i=0}^{k} p_i$
- 类别C_2概率为$P_2(k) = 1 - P_1(k)$
- 类别C_1的均值$m_1(k) = \frac{1}{P_1(k)} \sum_{i=0}^{k} ip_i$
- 类别C_2的均值$m_2(k) = \frac{1}{P_2(k)} \sum_{i=k+1}^{L-1} ip_i$
- 全局均值$m_G = \sum_{i=0}^{L-1} ip_i$
- 类间方差$\sigma_B{}^2 = P_1(k)(m_1(k) - m_G)^2 + P_2(k)(m_2(k) - m_G)^2$





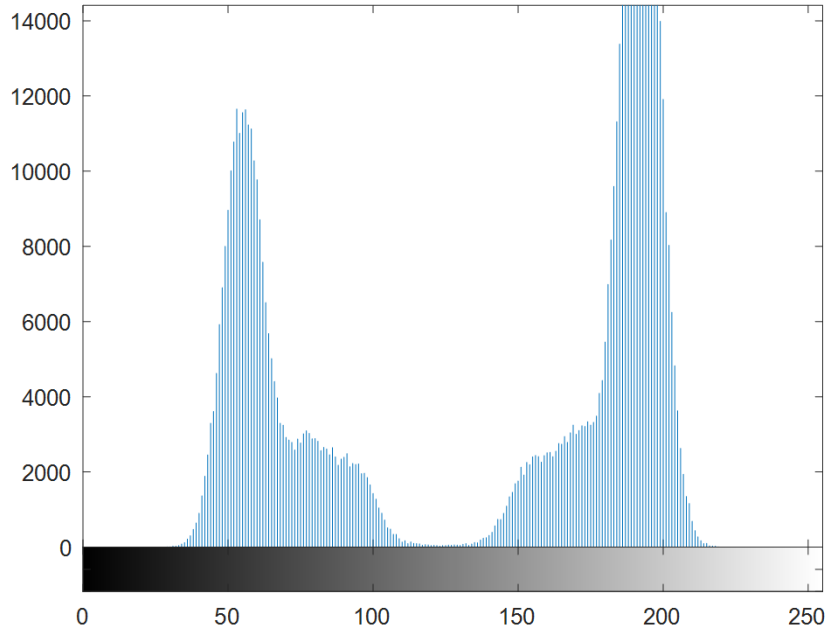
Otsu阈值法

计算各阈值下的类间方差，
选出最大值对应的阈值

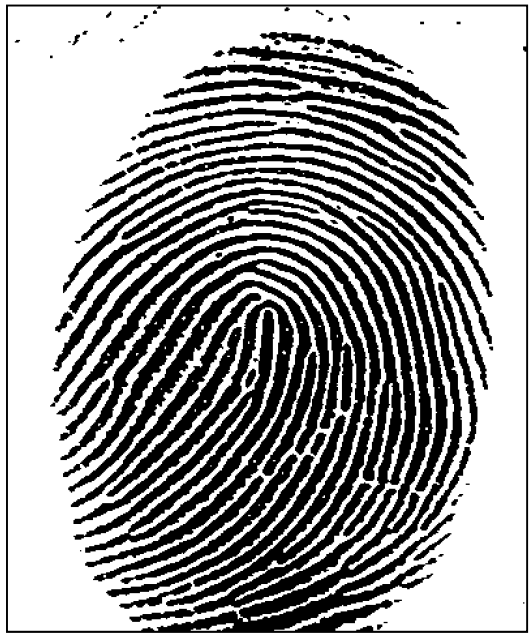
Threshold	T=0	T=1	T=2	T=3	T=4	T=5
	0	1.5928	2.5635	2.6287	2.1417	0.8705

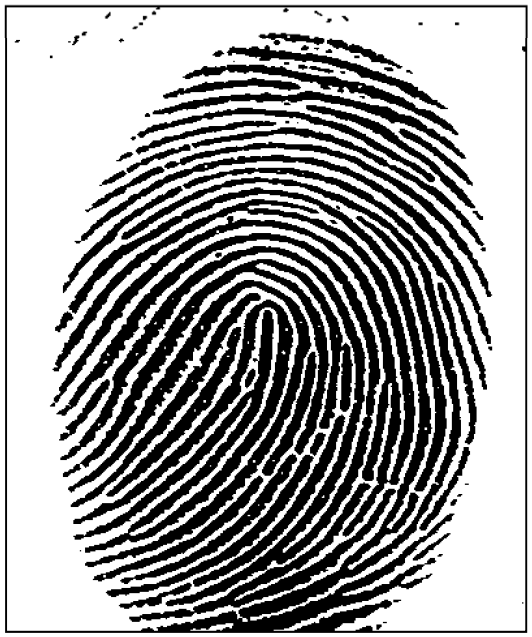
http://www.labbookpages.co.uk/software/imgProc/otsuThreshold.html

Otsu阈值法的例子



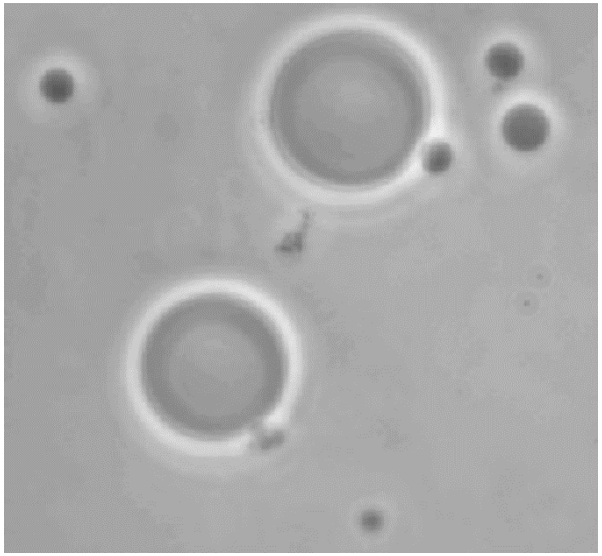
基本方法

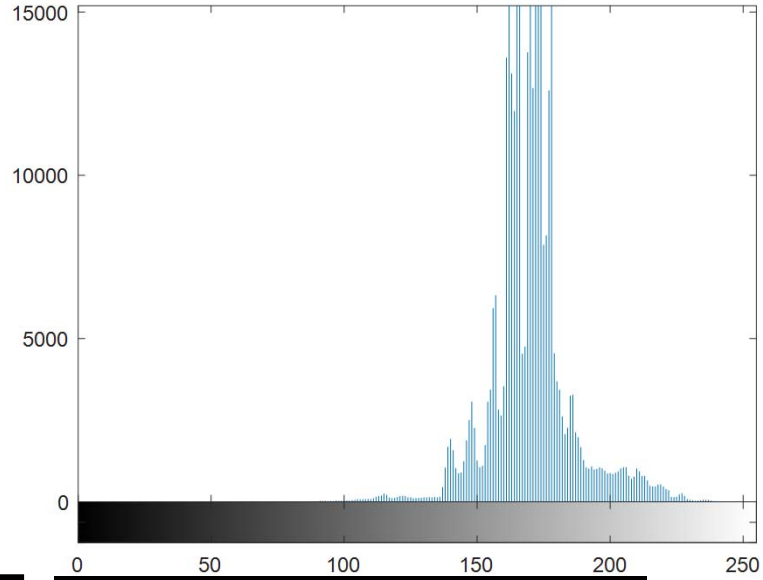




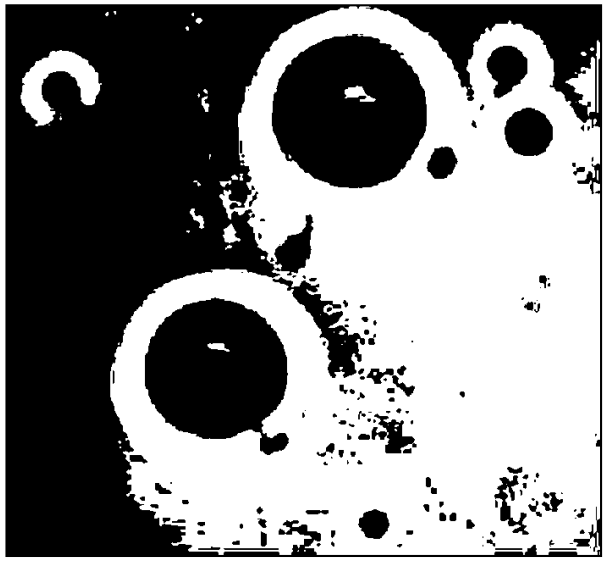
Otsu方法

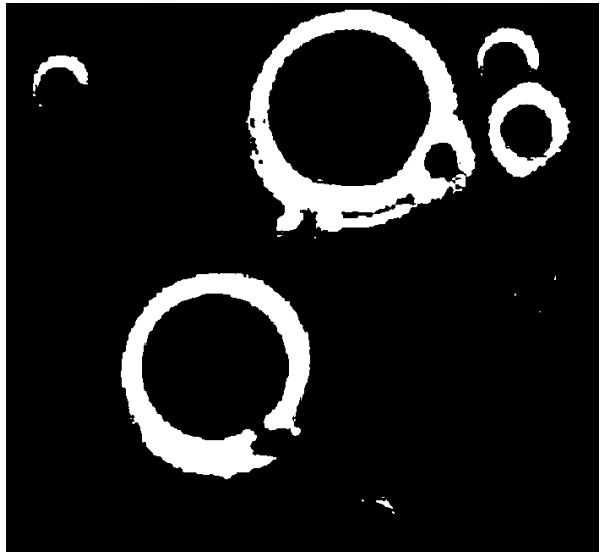
Otsu阈值法的例子





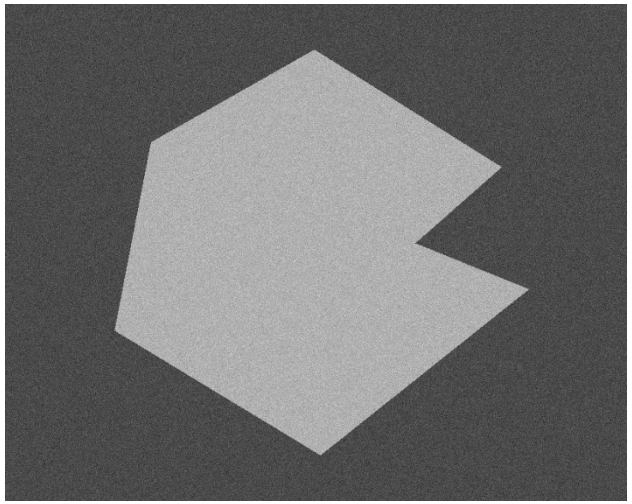
基本方法

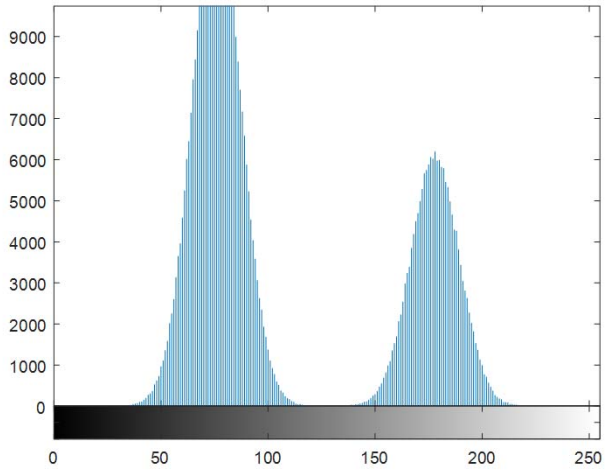


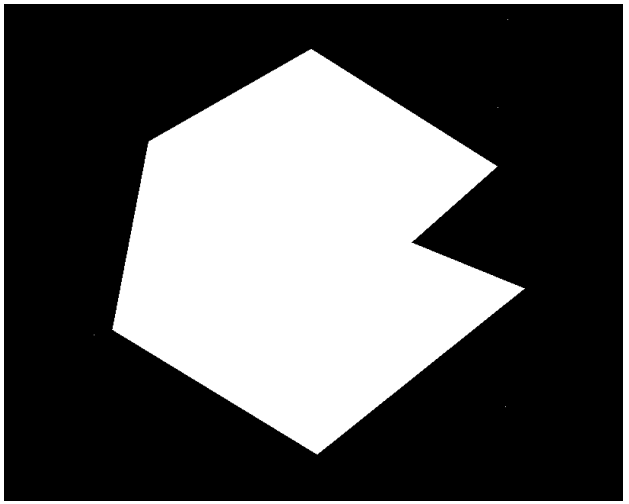


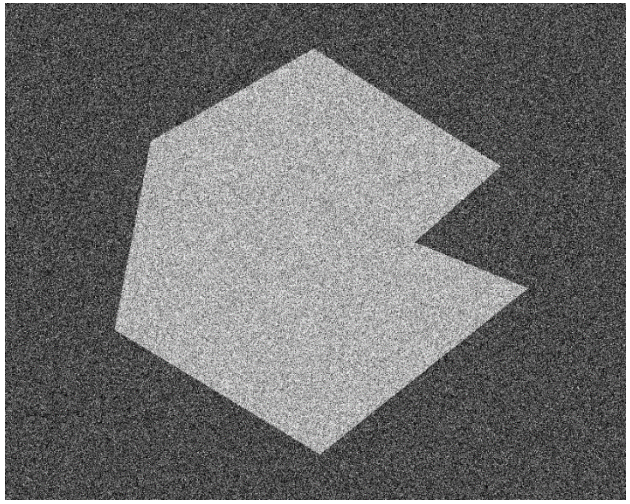
Otsu方法

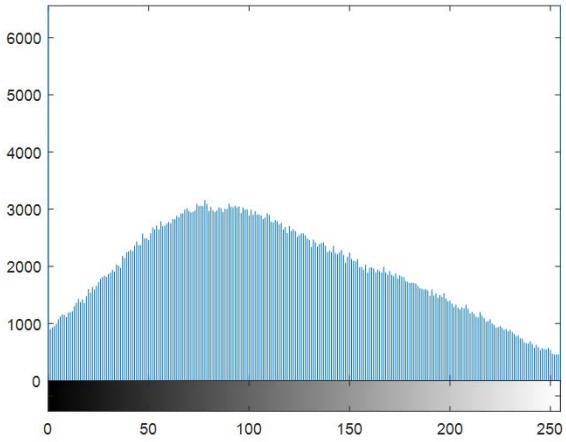
Otsu阈值法的例子

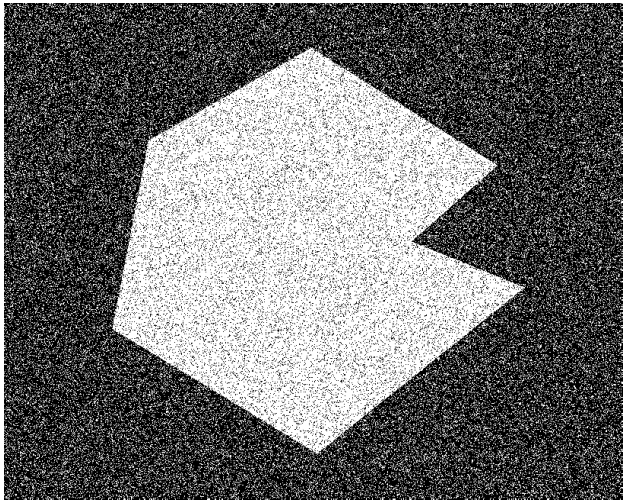






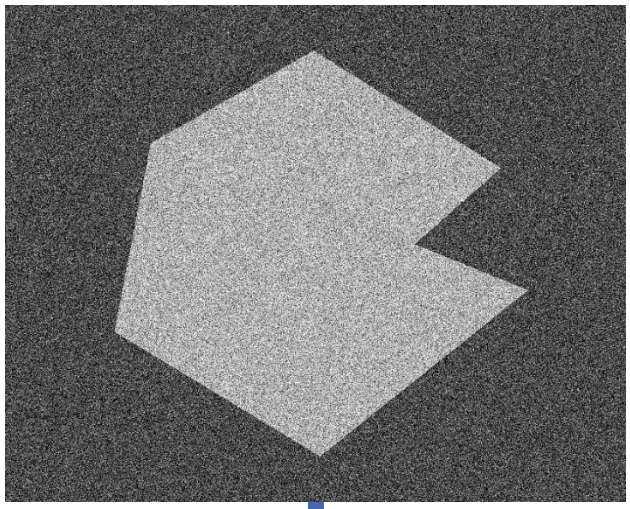


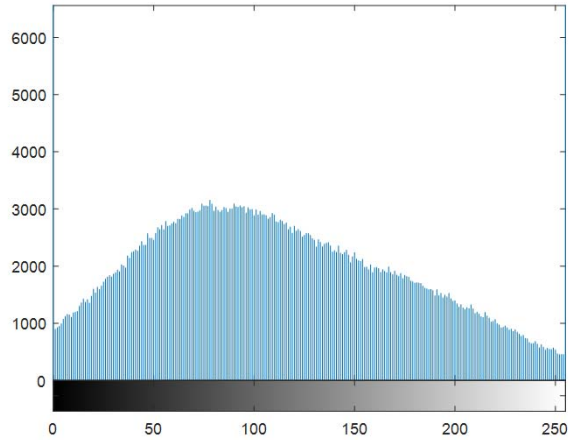


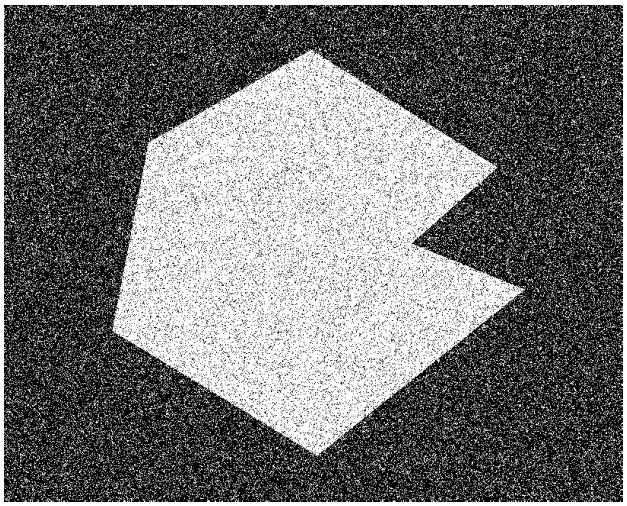


阈值法适合双峰分布（bimode distribution）的灰度直方图

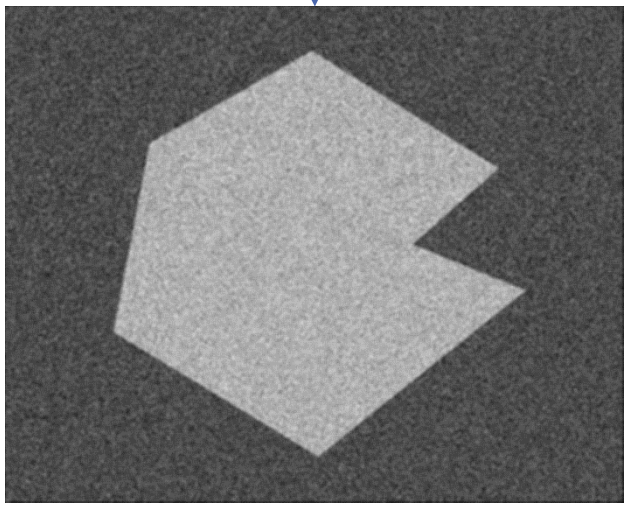
通过平滑改善阈值法

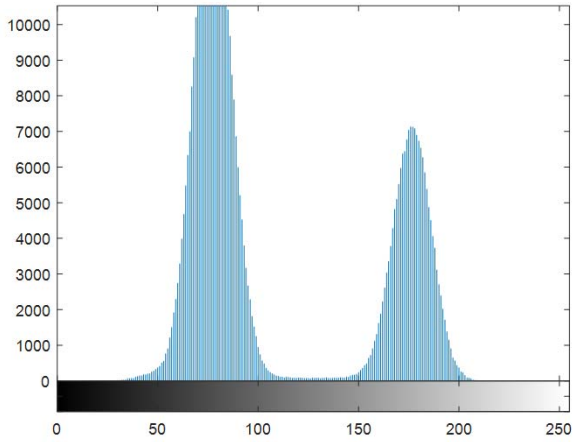






平滑





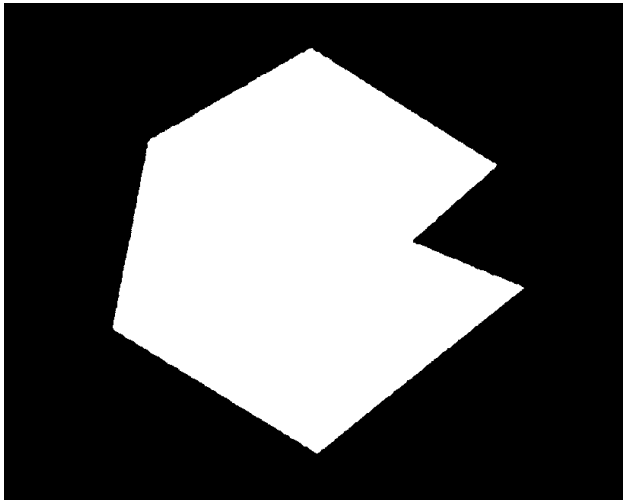
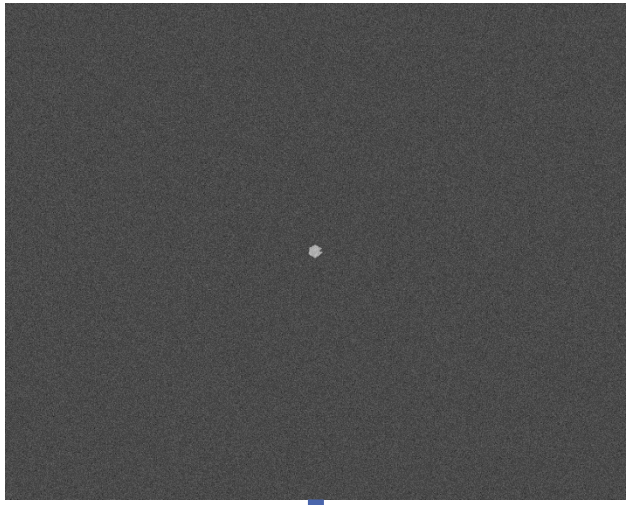
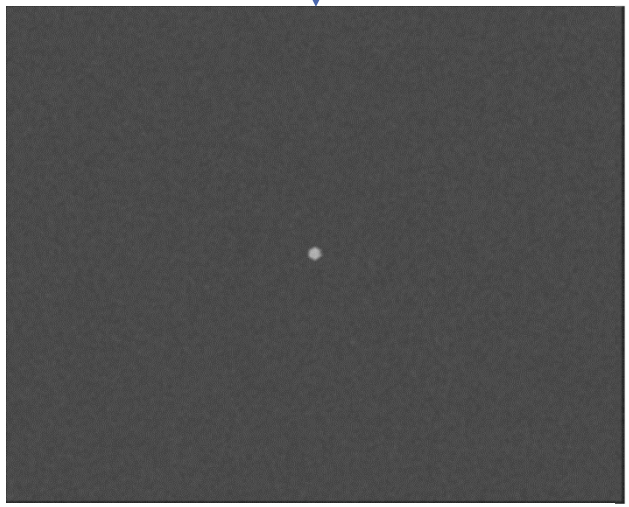


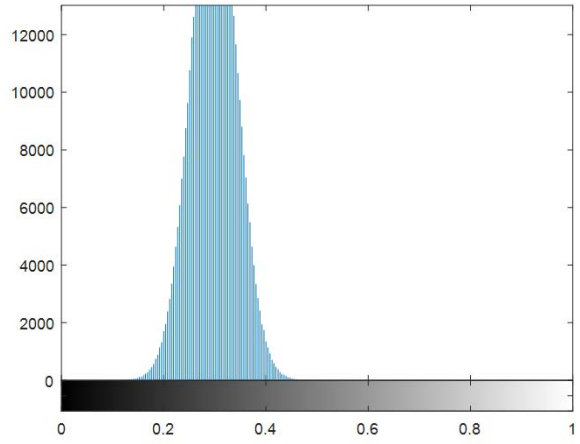
fig1040_smooth.m

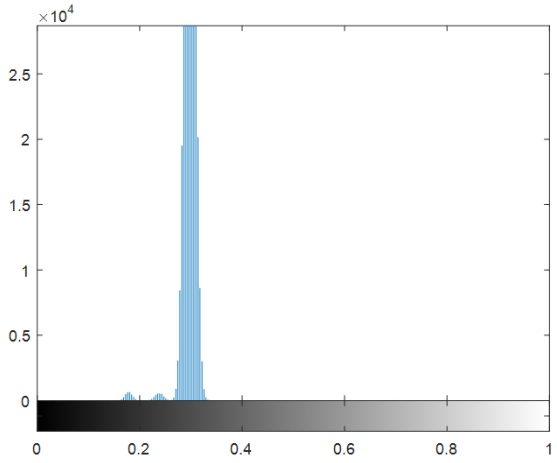
通过平滑改善阈值法

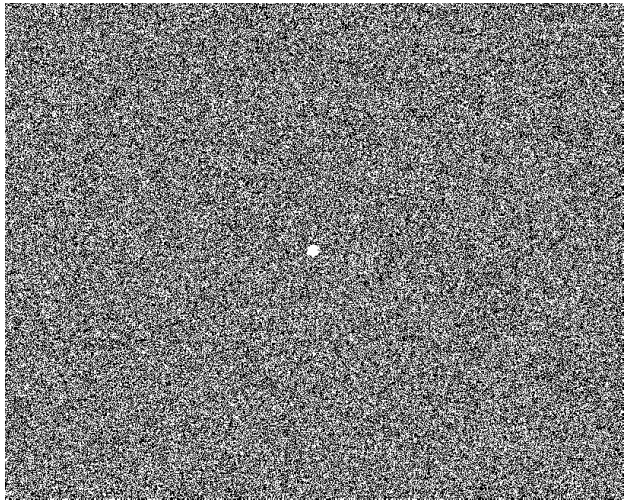


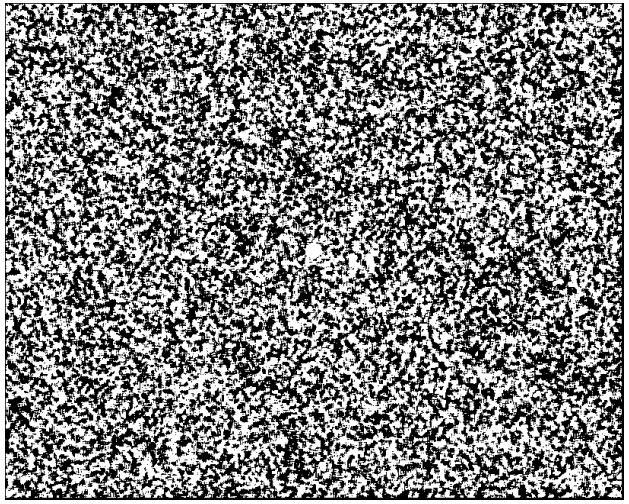
平滑







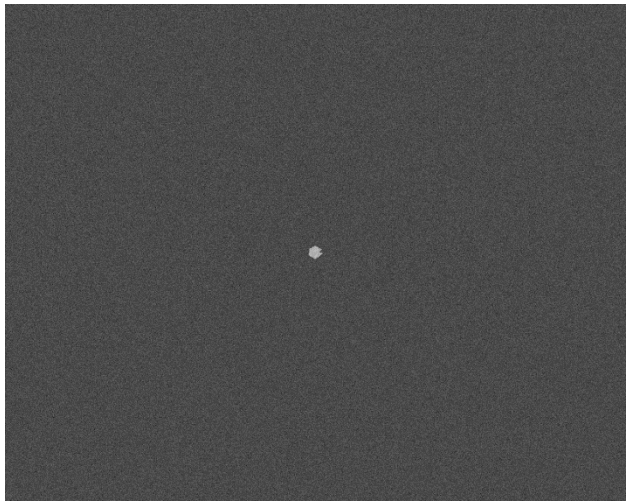




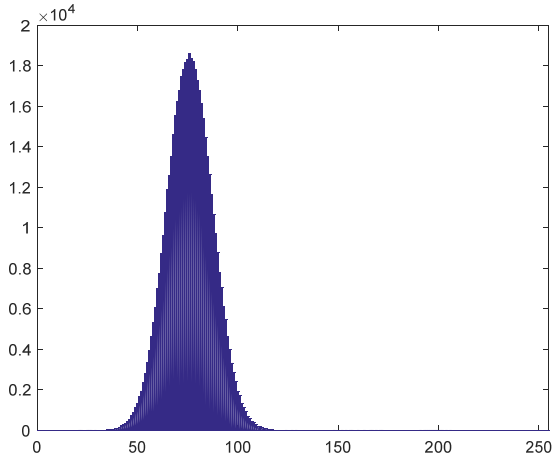
利用边缘改善阈值法

1. 计算输入图像的边缘图（梯度幅度或拉普拉斯绝对值）
2. 选择合适阈值，将边缘图二值化
3. 计算边缘处像素的灰度直方图
4. 根据直方图计算阈值
5. 利用该阈值将输入图像二值化

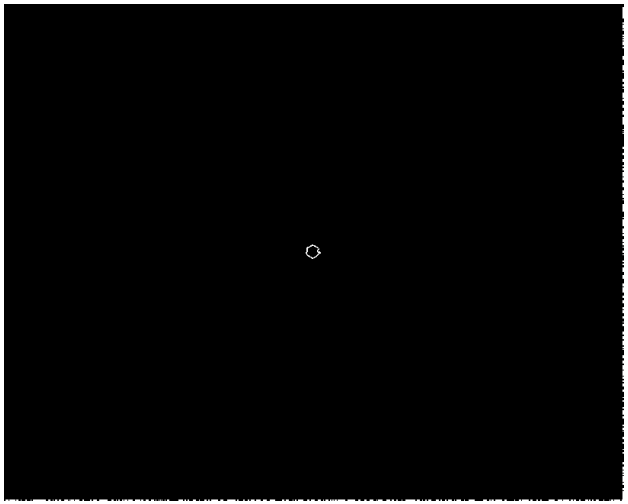
利用边缘改善阈值法



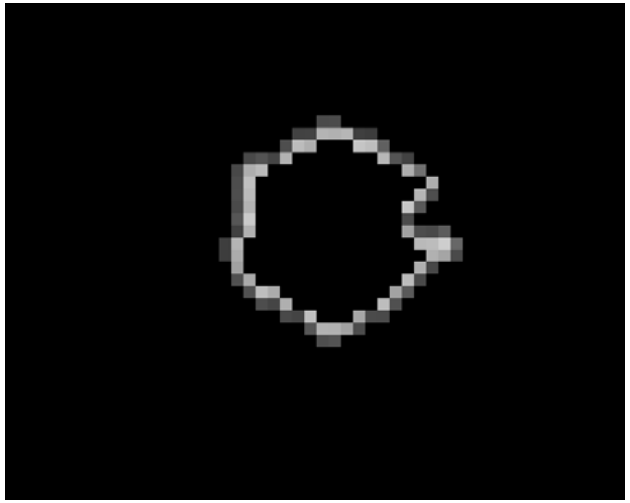
输入图像



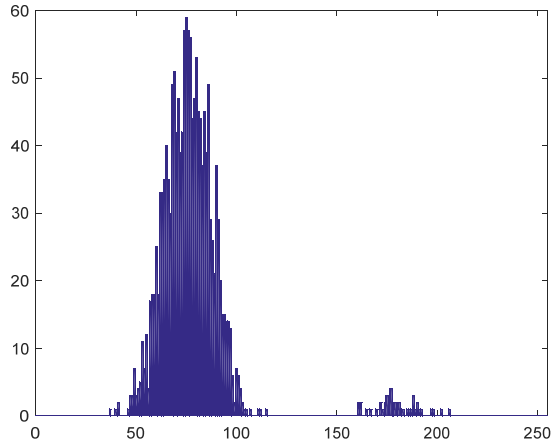
输入图像直方图



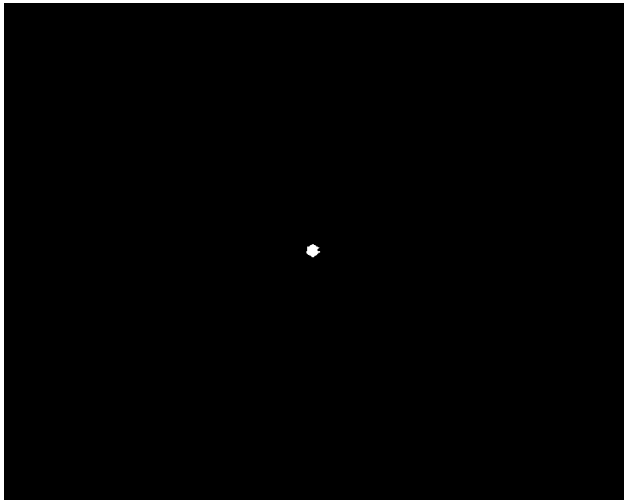
边缘图



局部放大图

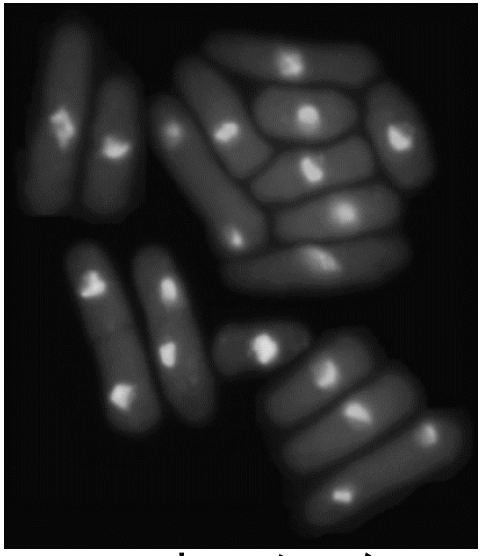


边缘像素的直方图

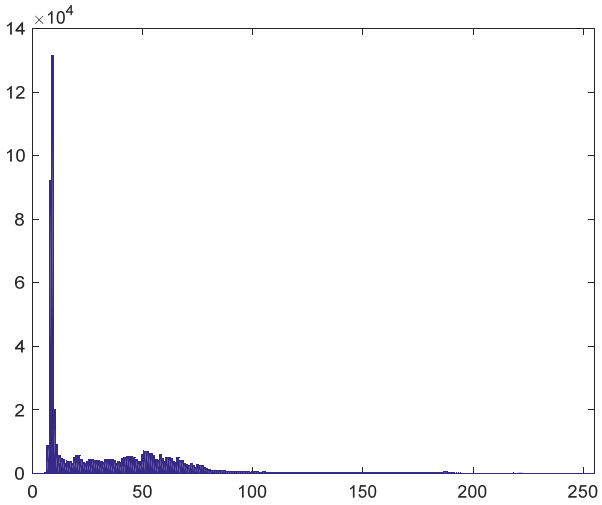


分割结果

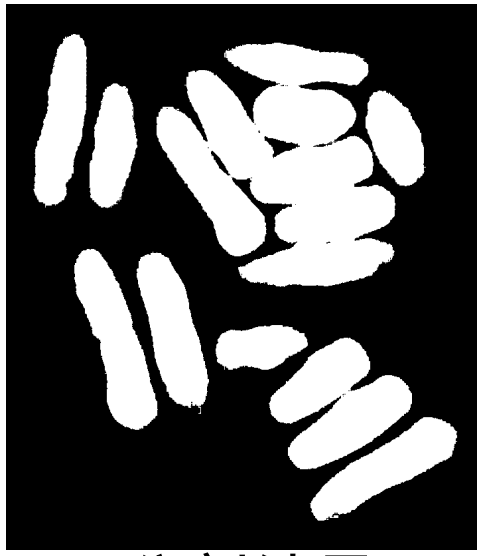
利用边缘改善阈值法



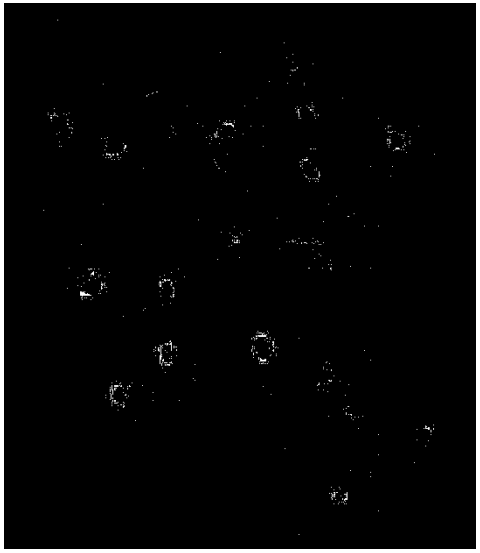
酵母细胞



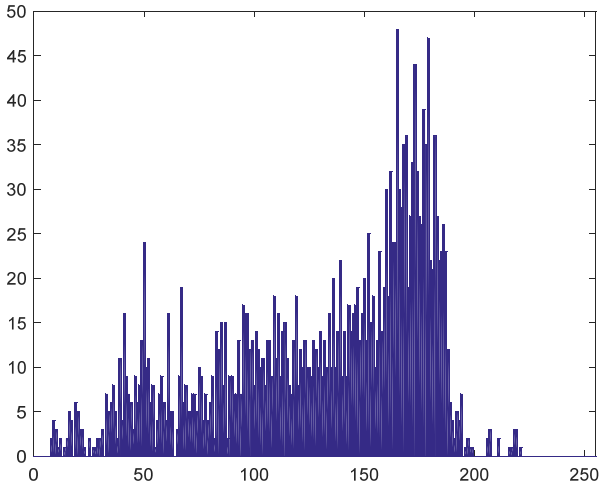
输入图像直方图



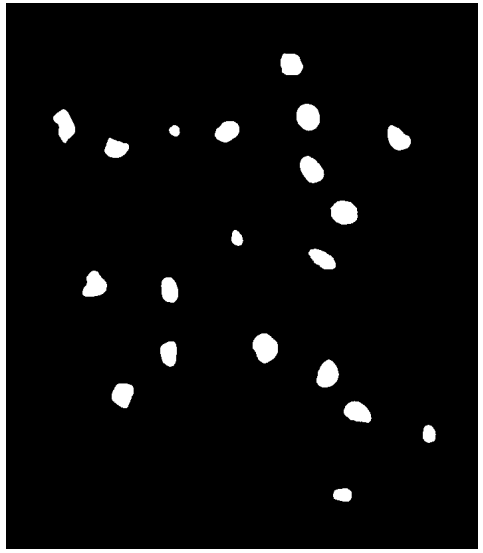
分割结果



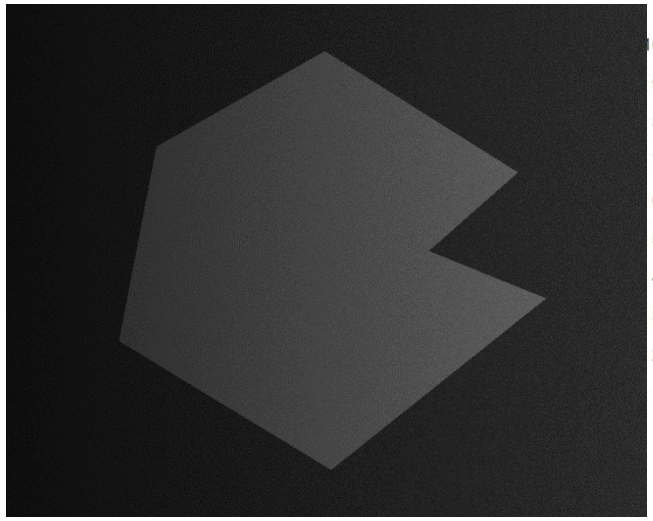
边缘图

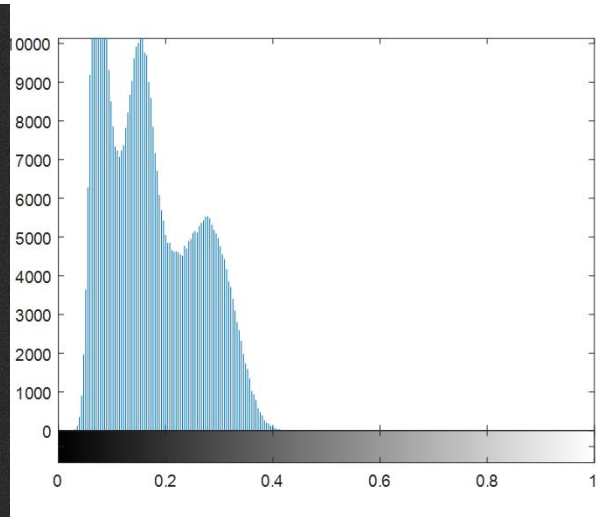


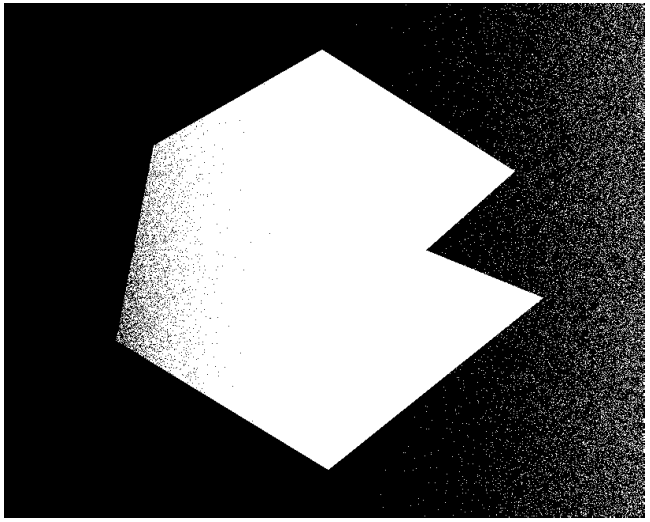
边缘像素的直方图



分割结果





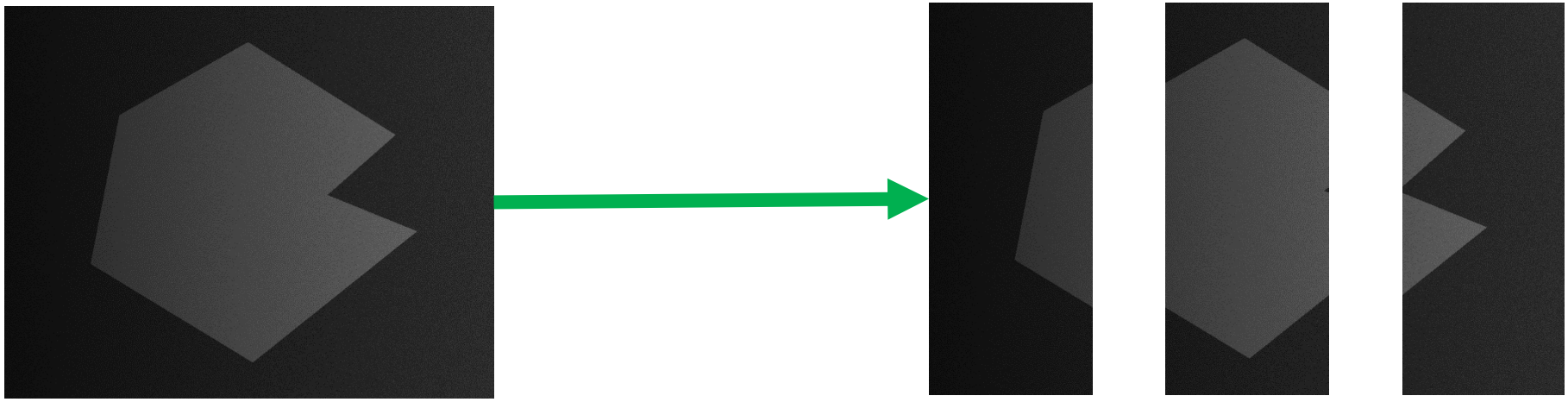


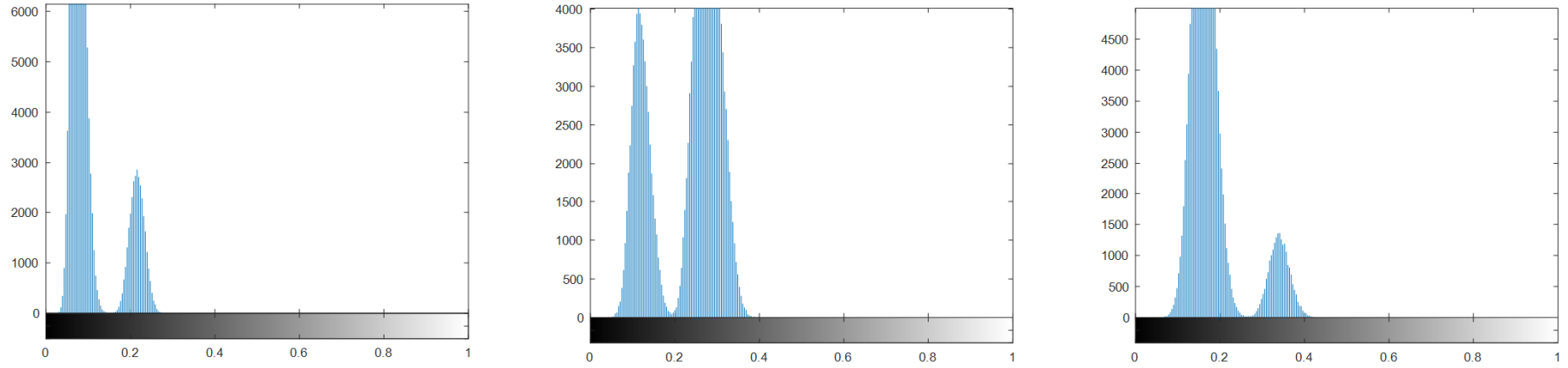
亮度不均匀导致全局阈值法失败

Top hat变换是一种解决办法

下面介绍另一种解决办法

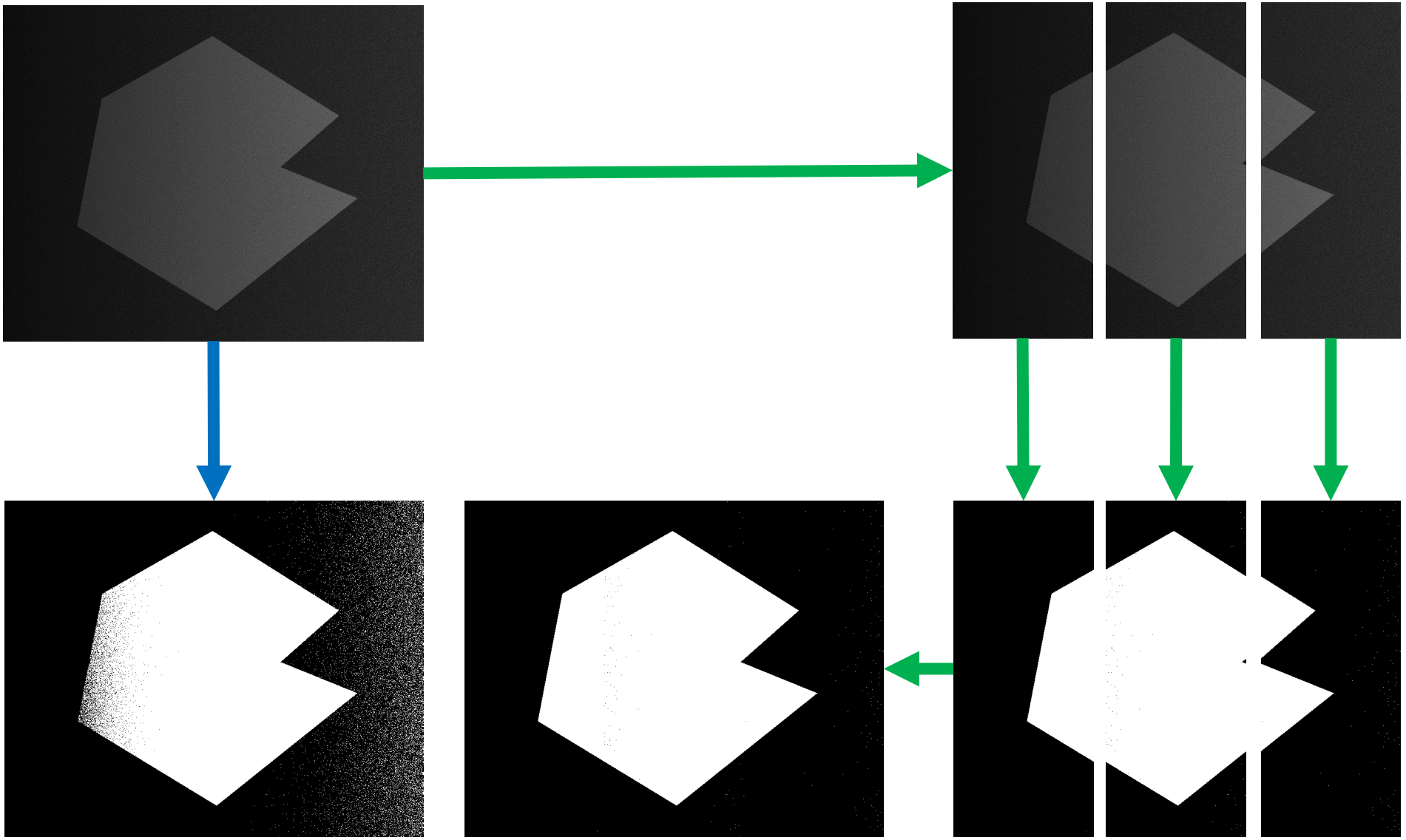
分区阈值

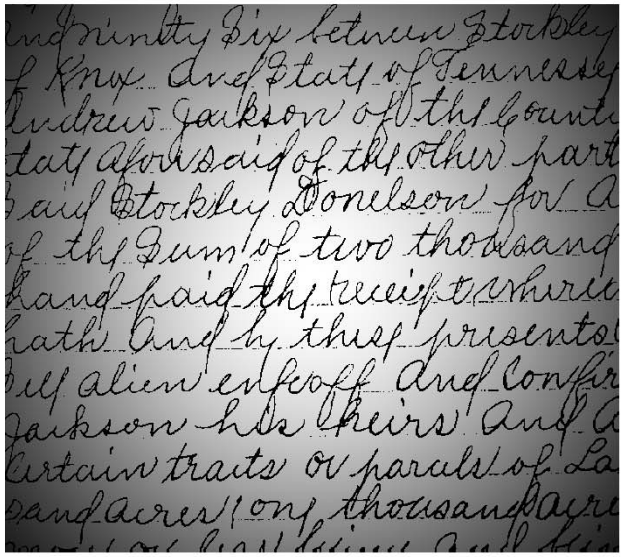


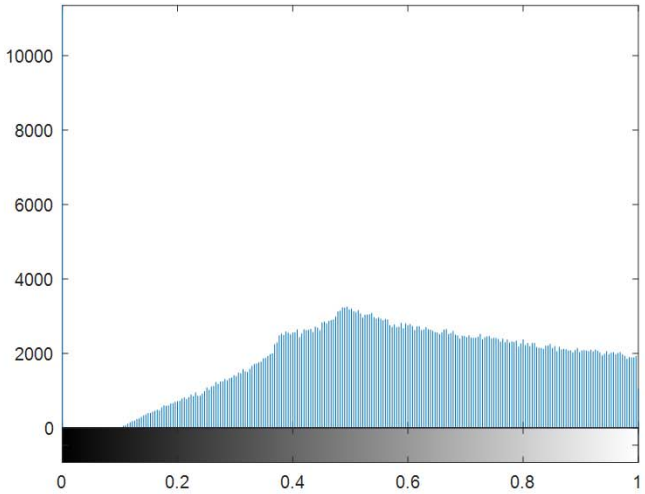


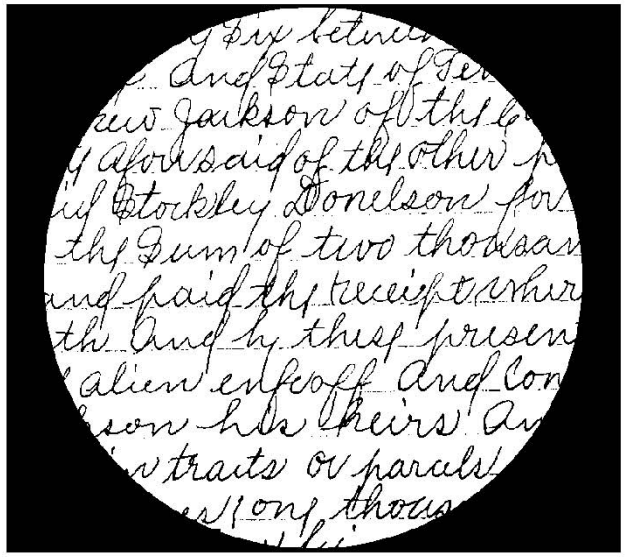
三幅子图像的直方图

分区阈值





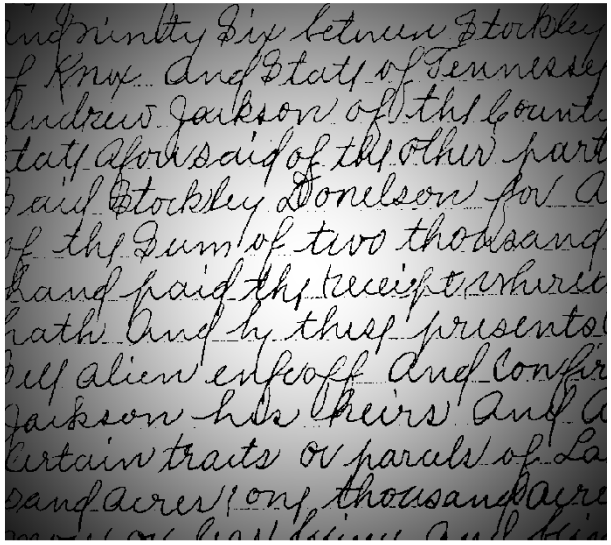


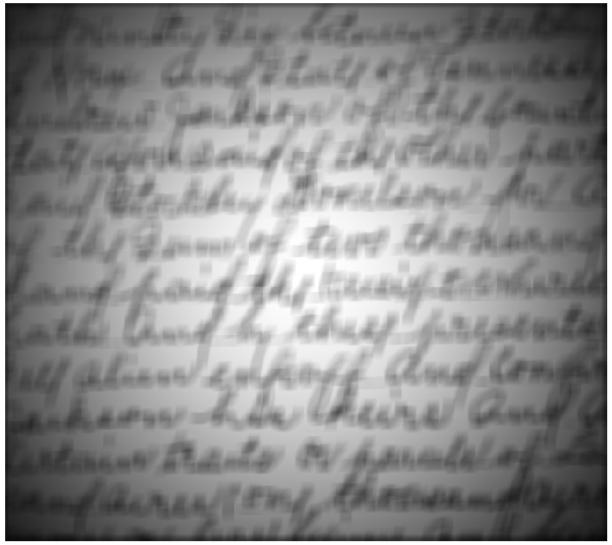


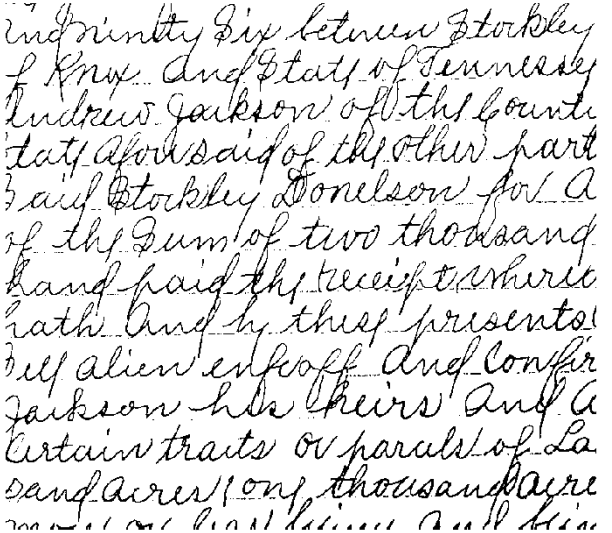
全局阈值失败

基于局部图像属性的阈值

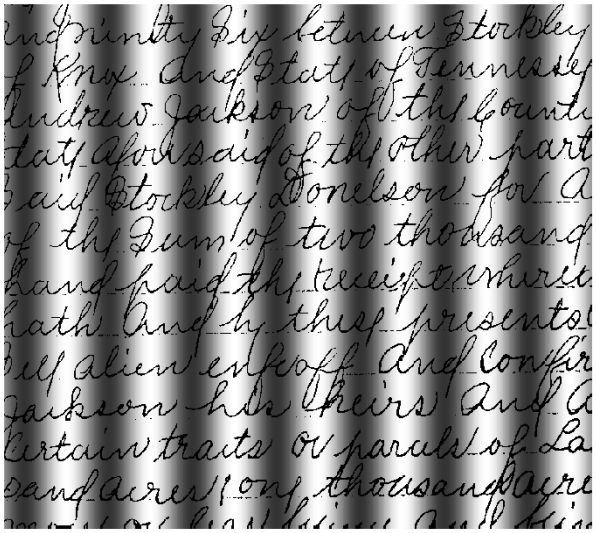
- 最常用的局部图像属性是局部均值
- 如果灰度大于局部均值则为1，否则为0

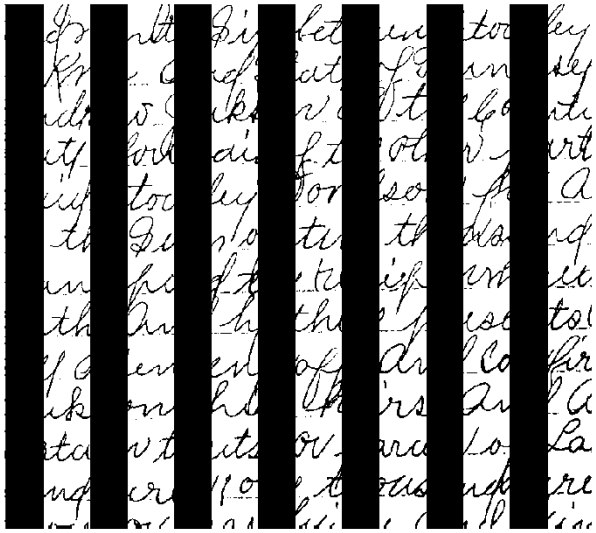




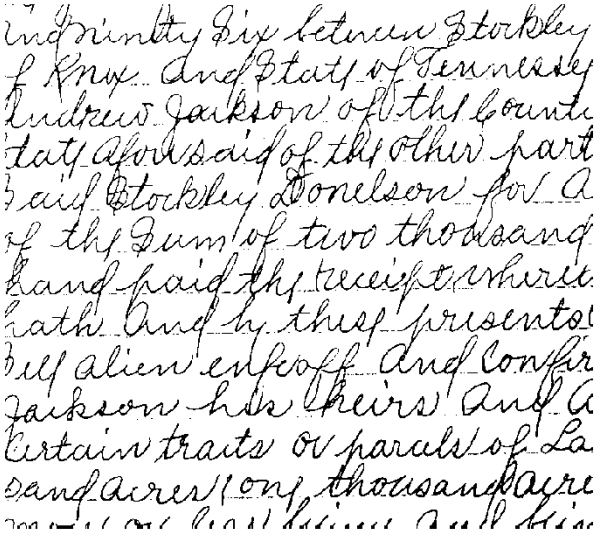


基于局部图像属性的阈值





全局阈值



局部阈值

内 容

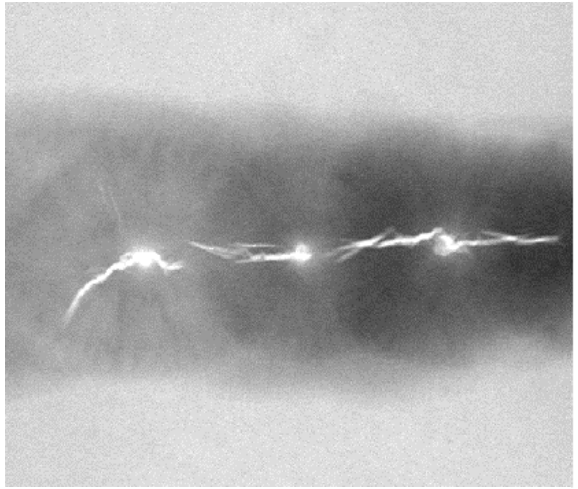
- 阈值法
- 基于区域的方法
- 分水岭法
- 其他方法

区域增长（Region Growing）

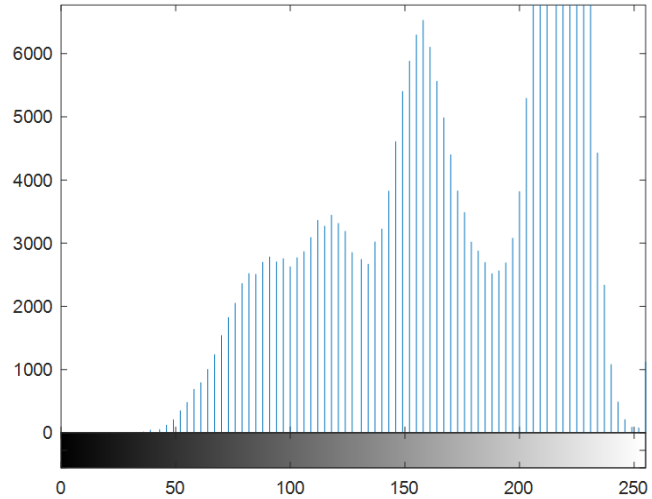
- 区域增长是一种经验性的方法，版本很多，没有普适的版本
- 下面介绍一个简单的版本：
  1. 找到种子点（手动或者用某种准则选择）
  2. 与种子点灰度值相似的像素为候选前景像素
  3. 将与种子点连通的候选前景像素加入各种子点所在的区域

区域增长的例子

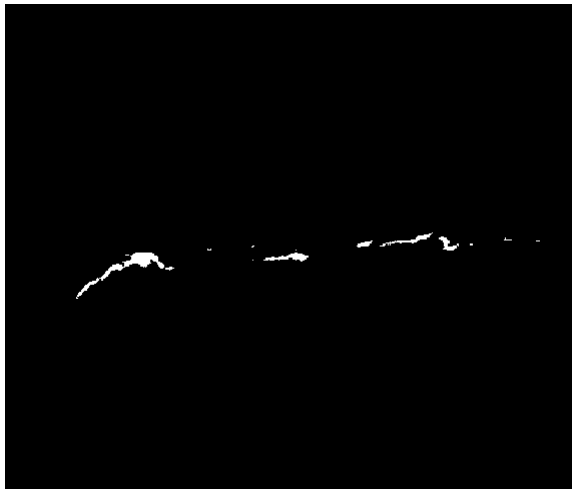
焊接质量X光检测



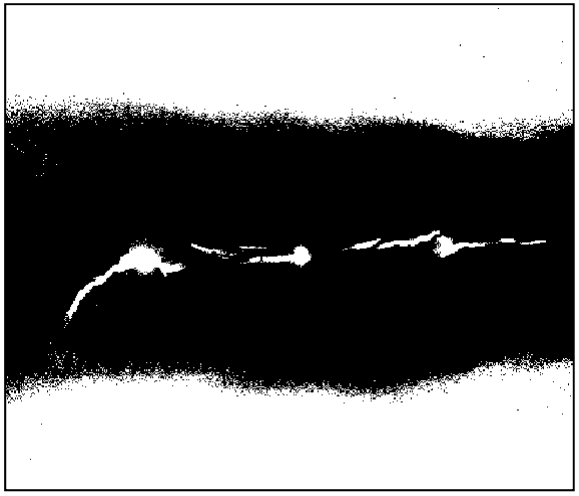
X光图像



X光图像的灰度值发图



种子点

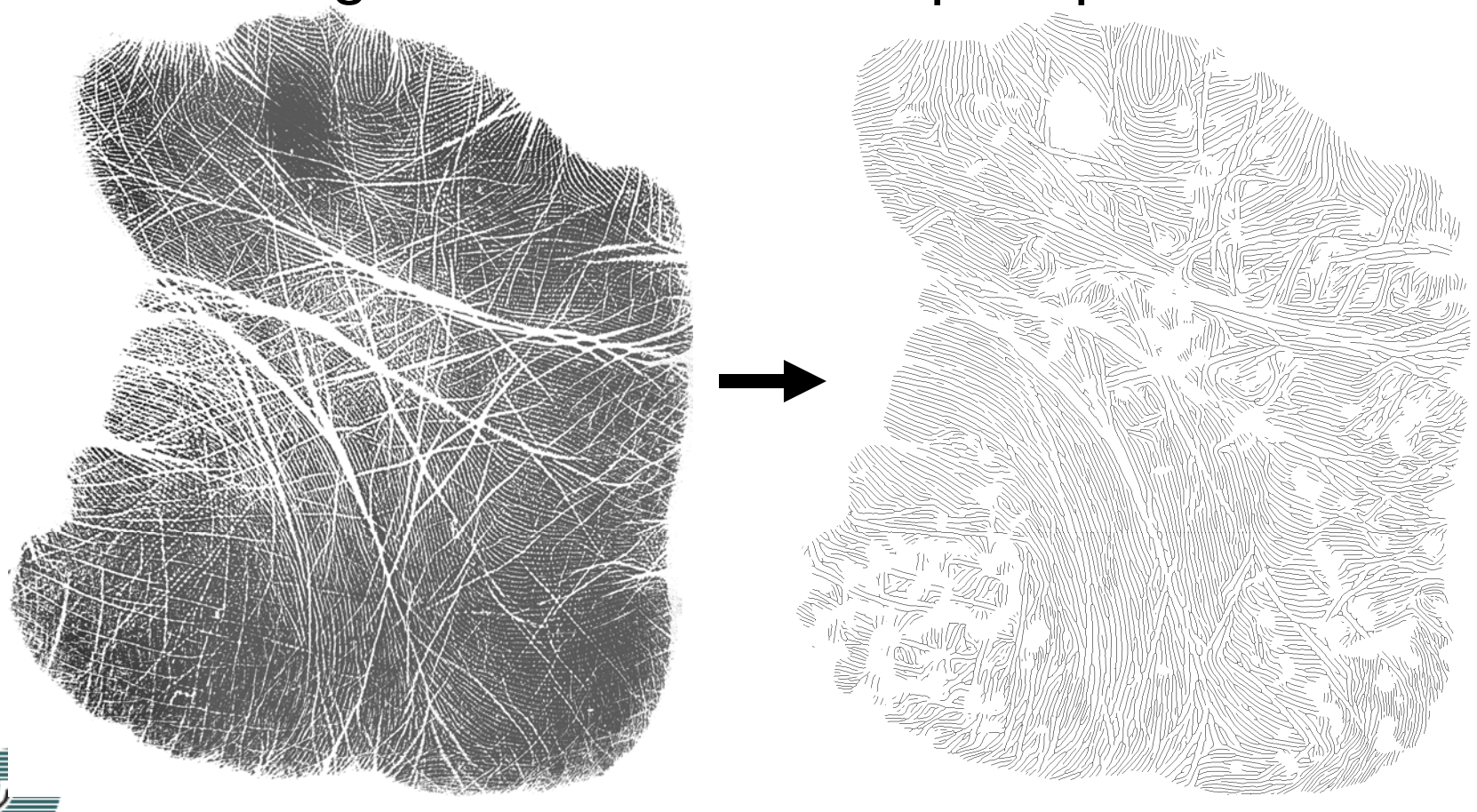


与种子点灰度值的差异小于68

区域增长结果

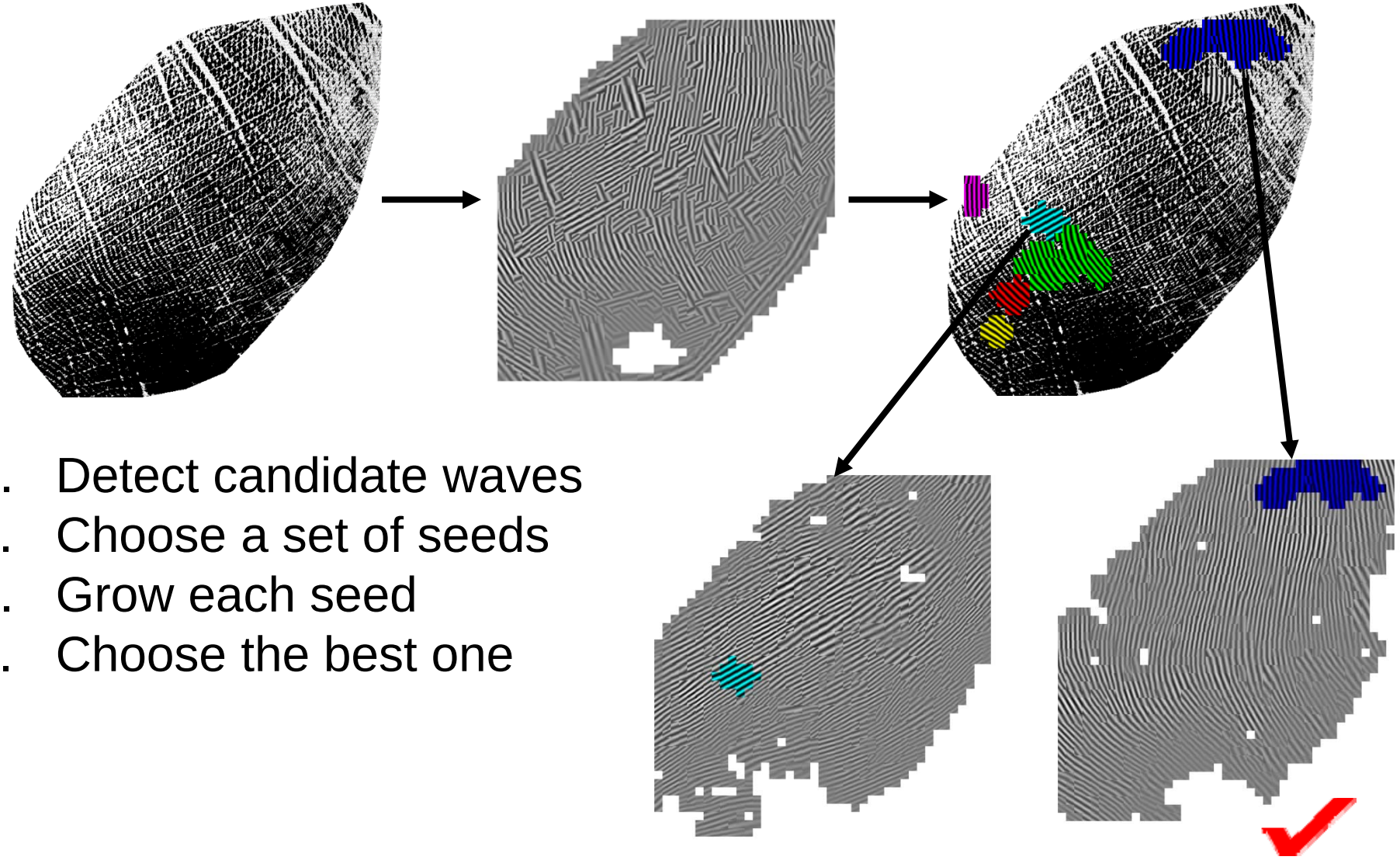
Challenge of palmprint

The large number of creases is the main challenge in estimating orientation field of palmprint.



NEUROtechnology
Biometric and Artificial Intelligence Technologies

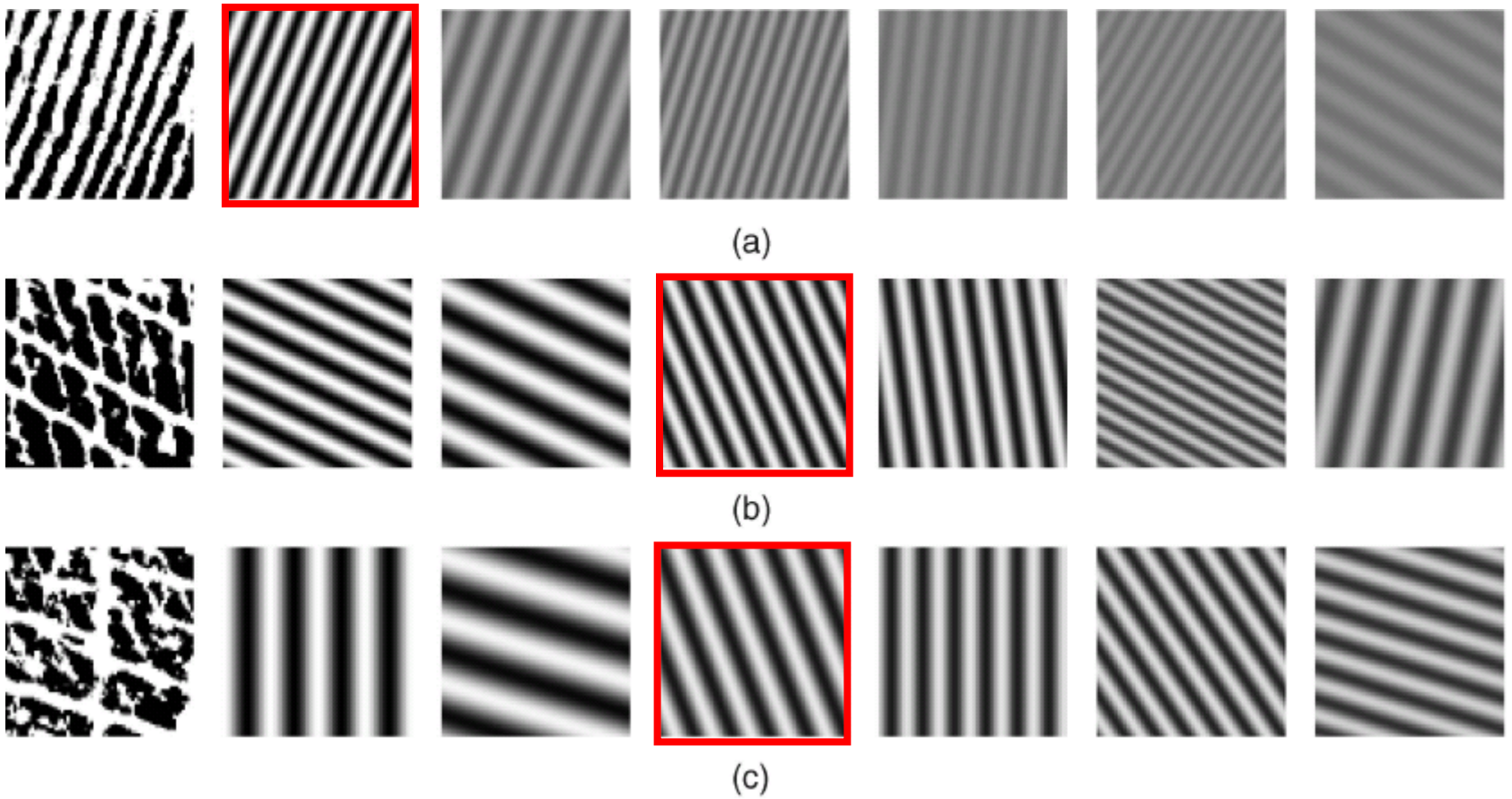
Steps of OF estimation



1. Detect candidate waves
2. Choose a set of seeds
3. Grow each seed
4. Choose the best one

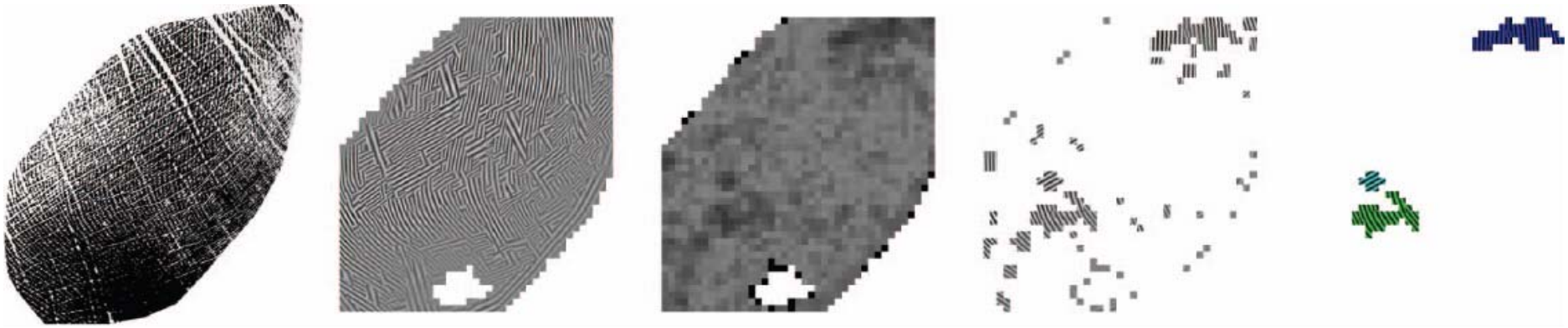
Anil K. Jain, Jianjiang Feng: Latent Palmprint Matching. IEEE T-PAMI 2009

Step 1: Candidate waves



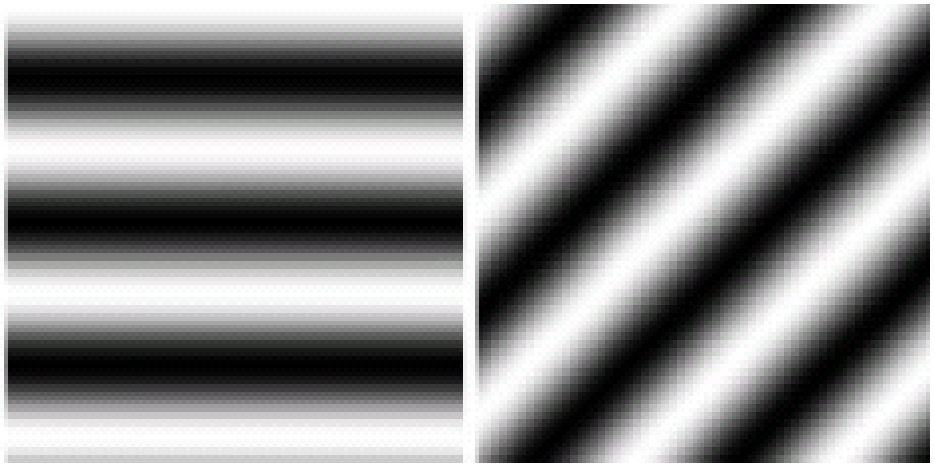
6 strongest sine waves corresponding to 3 types of local regions (64x64 pixels) in a palmprint: (a) no crease, (b) creases with one direction, (c) creases with two directions

Step 2: Seed selection

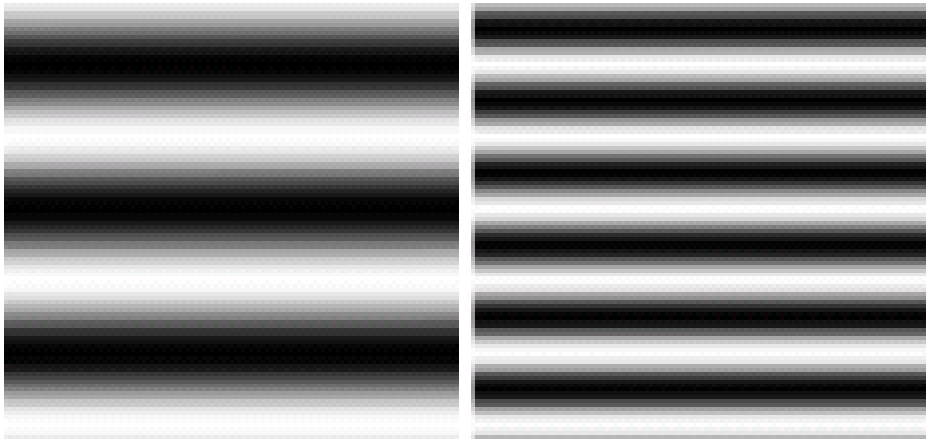


1. Compute the strongest sine waves in each block
2. If magnitude ratio is above a threshold, it is viewed as a reliable block
3. Reliable blocks compatible with each other are connected
4. Select the top N largest connected components as seeds

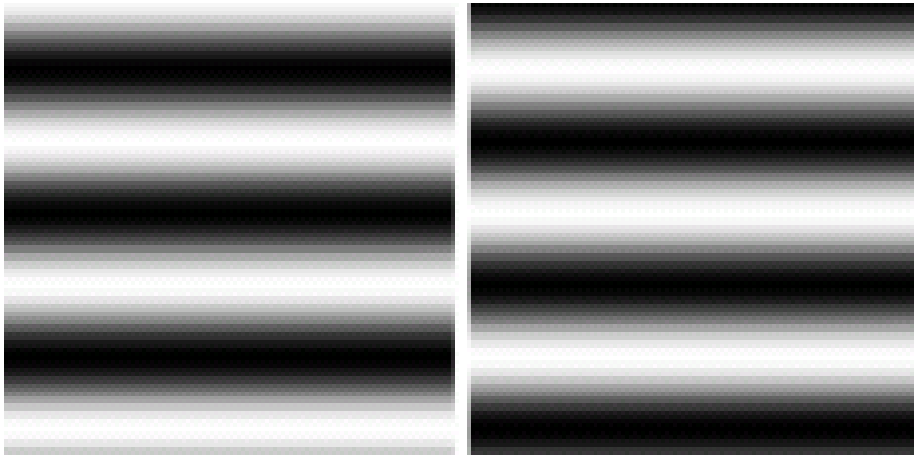
Step 2: Seed selection (compatibility)



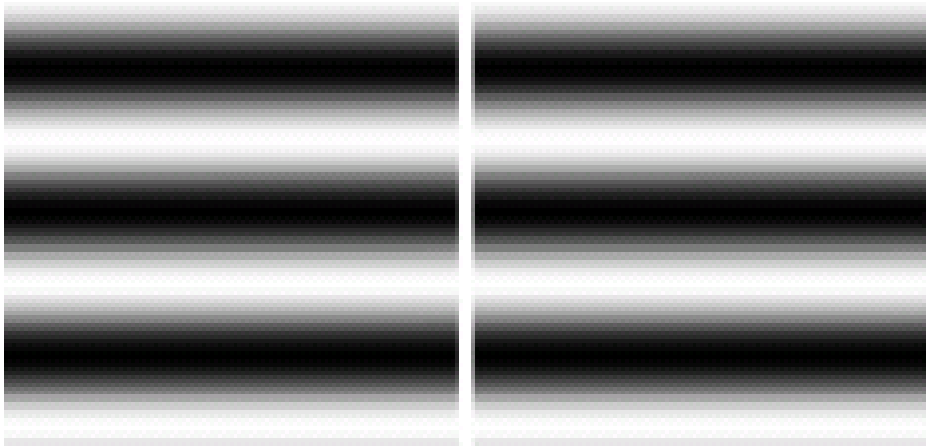
Incompatible orientation



Incompatible frequency

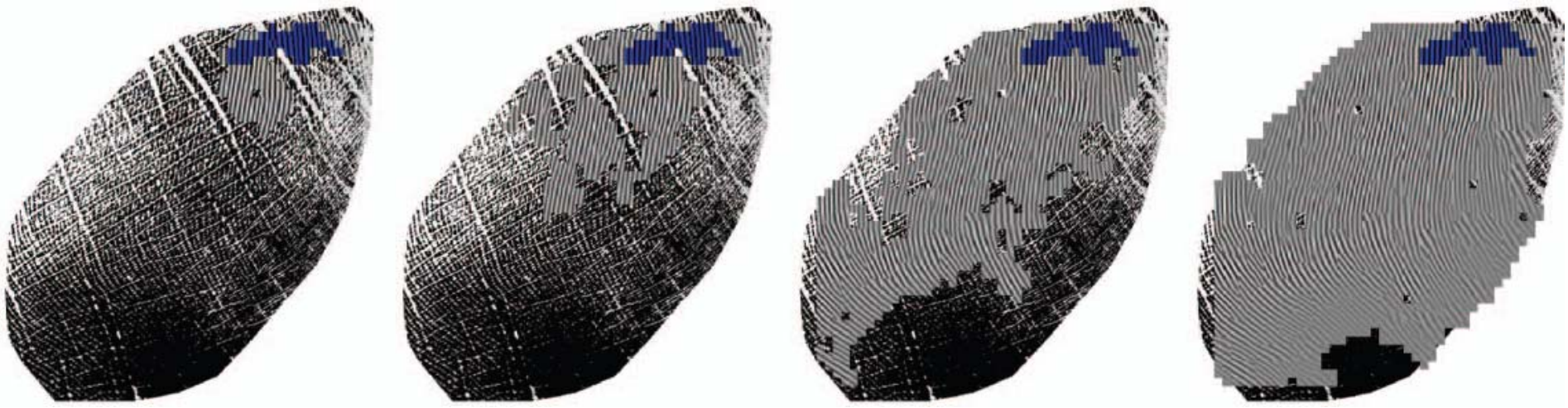


Incompatible phase



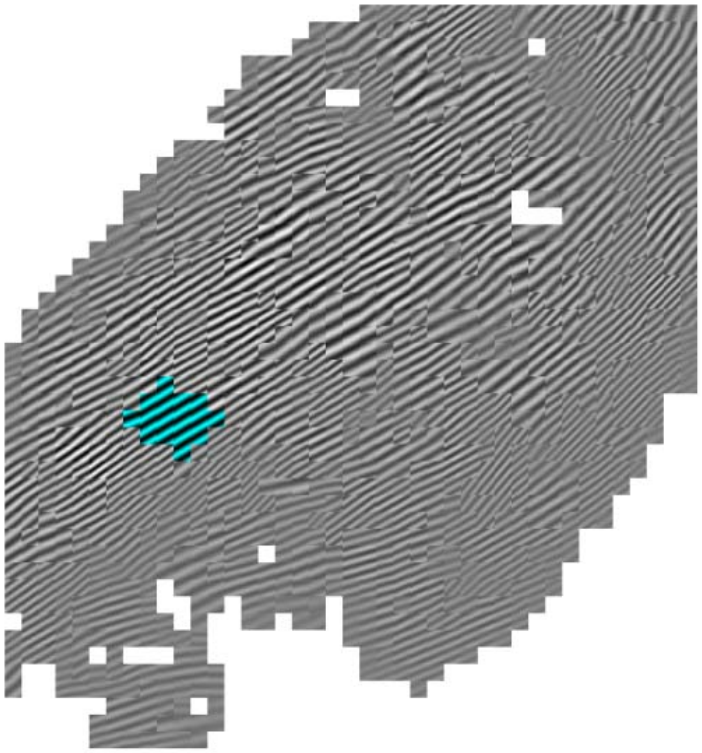
Compatible waves

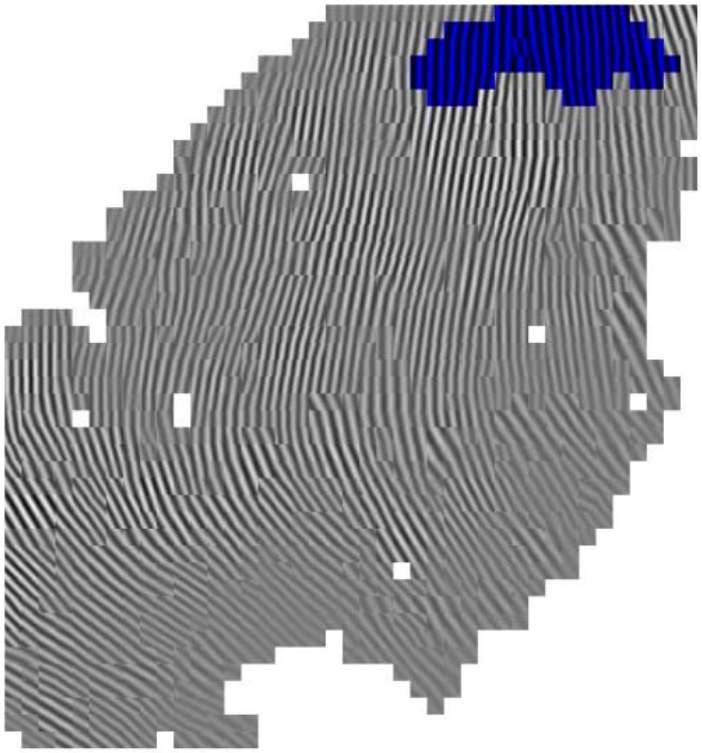
Step 3: Seed growing



Adjacent waves are greedily added. It is greedy because most compatible one is added first.

Step 4: Choose the best one





The number of reliable waves is counted. The pattern with largest number of reliable waves is chosen.

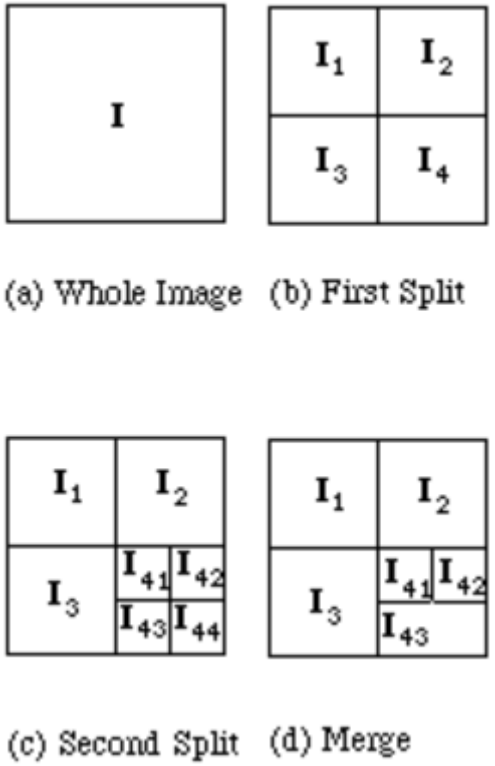
Comparison with VeriFinger

NEUROtechnology
Biometric and Artificial Intelligence Technologies

Proposed

区域分裂与合并（Region Splitting & Merging）

- 与区域生长的过程相反
- 分裂合并准则：某种区域一致性准则
- 分裂：当某区域不符合该准则时，将该区域分裂成4 个相同大小的子区域
- 合并：当相邻的子区域满足一致性特征时则将它们合成个大区域
- 结束条件：当没有区域能分裂和合并时
- 适合用四叉树表示区域
- 区域分裂与合并是一种经验性的方法，版本很多



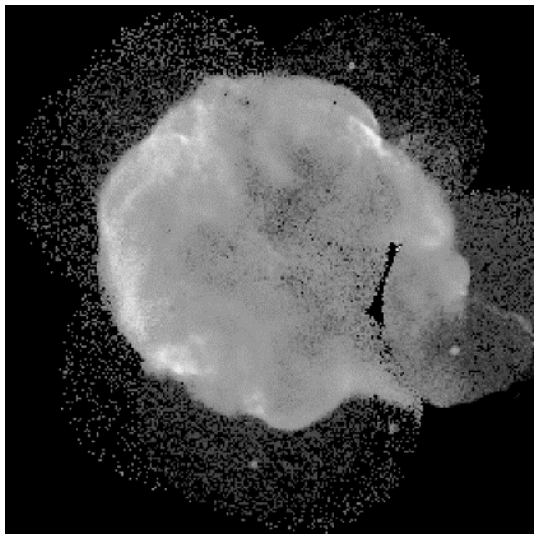
(a) Whole Image (b) First Split

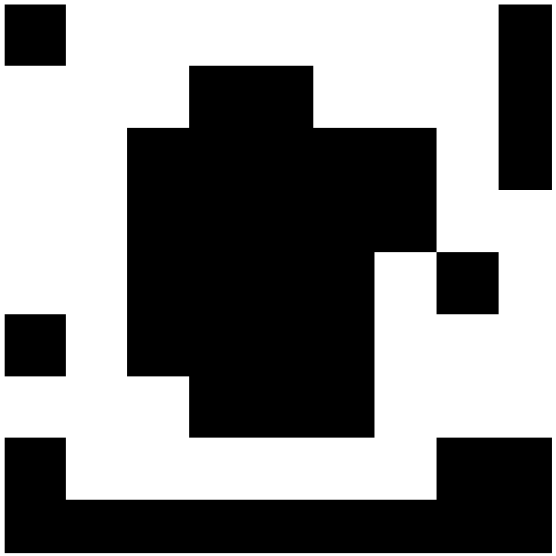
(c) Second Split (d) Merge

区域分裂与合并的例子

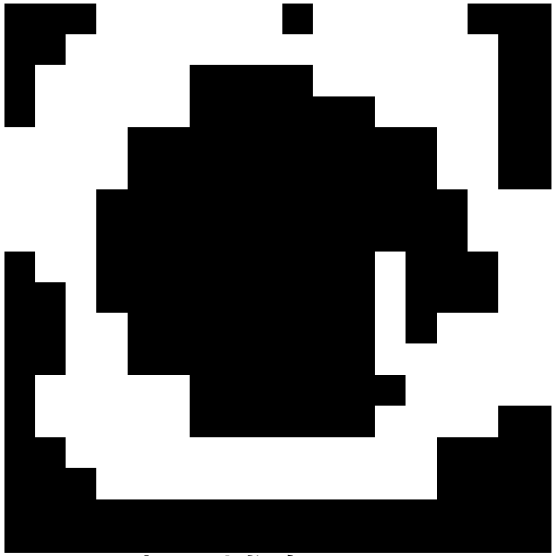
感兴趣区域
是周围环带

准则：方差$>a$
且0<均值$<b$





最小区域为64×64



最小区域为32×32

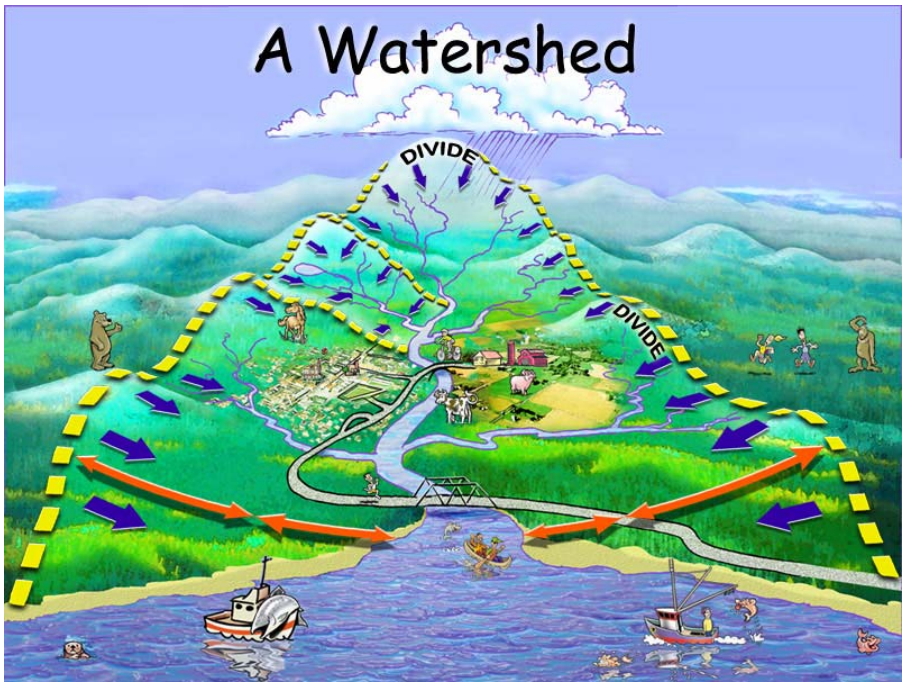
最小区域为16×16

最小区域为8×8

内 容

- 阈值法
- 基于区域的方法
- 分水岭法
- 其他方法

分水岭（Watershed）



- 三类点：

1. 极小值点
2. 滴到该点的水滴，总是会流到某极小值点
3. 滴到该点的水滴，有可能流到多个极小值点

- 对于某极小值点，符合条件2的点构成其集水盆地（catchment basin，watershed），符合条件3的点构成其分水线（divide line, watershed line, ridge line）

分水岭分割算法

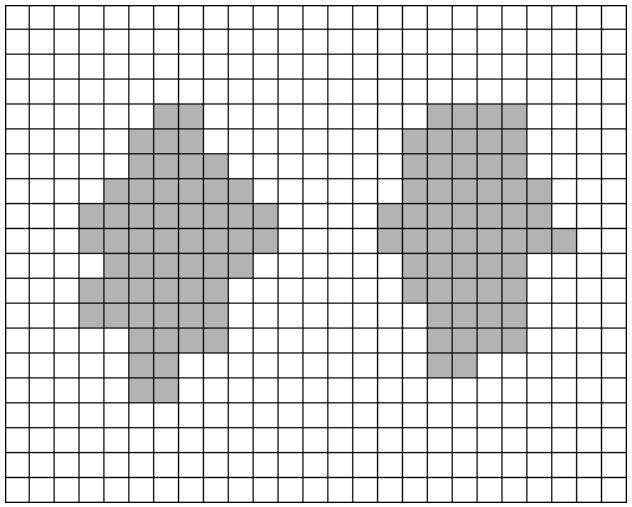
- 将图像灰度视为高度，将图像视为地形图
- 基本思想：

在局部极小值点处戳洞，从洞口往上以恒定速度注水。当两个不同集水盆地的水要合并时，修水坝（高度超过最高的山）防止合并。继续注入水，直到水位达到最高的山顶。这时，水面上只有水坝，这些就是算法找到的边界。

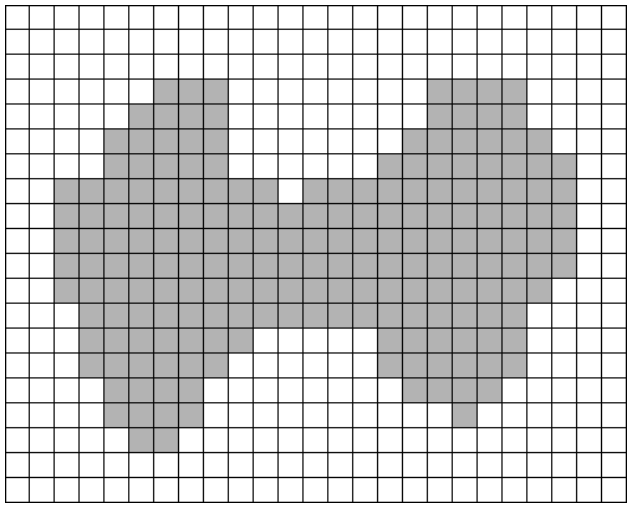
- 分水岭算法的动画：
  http://cmm.ensmp.fr/~beucher/wtshed.html

如何修水坝

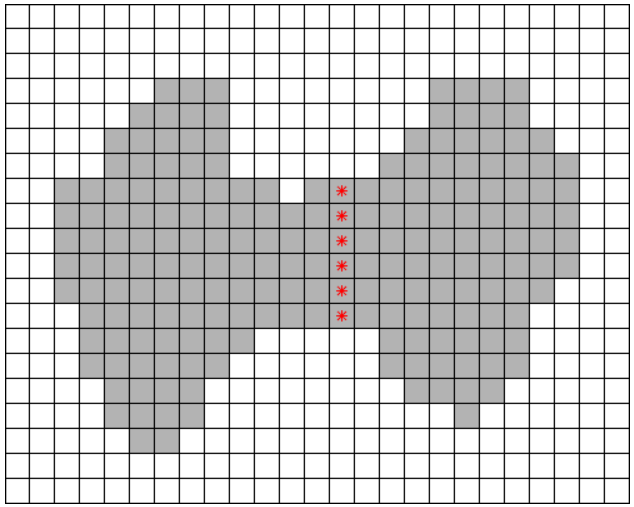
水坝（dam）即算法要找的分水岭线
通过迭代的形态学膨胀找水坝



水位为$n-1$时，
被部分淹没的两个
盆地



水位为n时，两个
盆地连通



水坝

分水岭算法

- 令$M_1, M_2, \ldots, M_R$表示图像$g(x,y)$极小值点坐标的集合
- 令$C(M_i)$表示极小值点集合M_i的集水盆地
- 令$T[n]$表示像素值小于n的坐标集合，即当水位线高n时，图像中被淹没的坐标集合
- 令$C_n(M_i)$表示当水位线高n时，$C(M_i)$中被淹没的坐标集合
- $C_n(M_i) = C(M_i) \cap T[n]$
- 令$C[n] = \cup_{i=1}^{R} C_n(M_i)$
- $C[\max + 1] = \cup_{i=1}^{R} C(M_i)$，全部集水盆地的集合

分水岭算法

- 分水岭算法通常用于梯度图，而非原始图像
- 梯度图中极小值对应原始图像的平坦区，而边缘将这些平坦区隔开
- 分水岭算法找到的边界正好对应边缘

内 容

- 阈值法
- 基于区域的方法
- 分水岭法
- 其他方法