机器视觉与图像处理

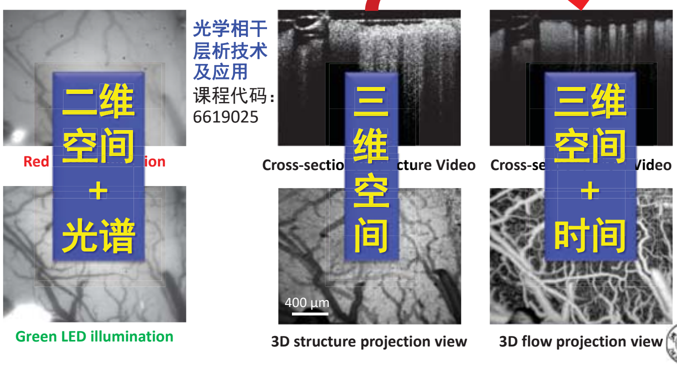
第8讲 形态学图像处理

李 鹏
光电科学与工程学院，玉泉，教三-311
Email: peng_li@zju.edu.cn
HomePage: http://person.zju.edu.cn/lipeng

回顾

- 第1讲 绪论
  – 让“机器”像人一样识别物品、理解场景　　能看见 + 能理解
- 第2讲 图像的获取　输入端
  – 照明、镜头、图像传感器，“好的图像成功一半”
- 第3讲 图像的基础变换
  – 点处理及灰度直方图、代数变换、几何变换
- 第4讲 图像的空间域增强
  – 图像的平滑、图像中值滤波、图像锐化，相关与卷积
- 第5讲 图像的频率域增强
  – 图像空域到频域的转换，变换结果的理解，频域滤波（低通高通带通带阻）
- 第6讲 图像的退化与复原
  – 图像的退化模型，图像退化与噪声模型的判断，滤波复原（空域与频域）、逆滤波复原
- 第7讲 彩色图像与高光谱图像
  – 彩色图像模型，伪彩色增强，真彩色增强，RGB-HIS，颜色空间变换，多光谱图像的作用和价值

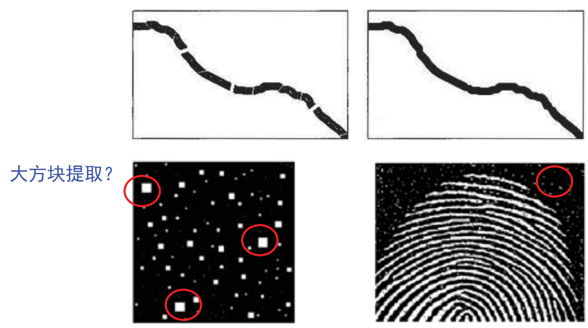
好的成像技术，事半功倍



回顾

- 第1讲 绪论 能看见+能理解
  – 让“机器”像人一样识别物品、理解场景
- 第2讲 图像的获取
  – 图像传感器、镜头、光照，“好的图像成功一半”
- 第3讲 图像的基础变换
  – 点处理及灰度直方图、代数变换、几何变换
- 第4讲 图像的空间域增强
  – 图像的平滑、图像中值滤波、图像锐化，相关与卷积
- 第5讲 图像的频率域增强
  – 图像空域到频域的转换，变换结果的理解，频域滤波（低通高通带通带阻）
- 第6讲 图像的退化与复原
  – 图像的退化模型，图像退化与噪声模型的判断，滤波复原（空域与频域）、逆滤波复原
- 第7讲 彩色图像与高光谱图像
  – 彩色图像模型，伪彩色增强，真彩色增强，RGB-HIS，颜色空间变换，多光谱图像的作用和价值

问题与动机



数学形态学Mathematical morphology

- 以形态为基础对图像进行分析的数学工具
- 基本思想：
  – 用具有一定形态的结构元素去度量和提取图像中的对应形状
  – 以达到对图像分析和识别的目的
- 集合论：形态学图像处理的数学基础和所用语言
  – 一种简单的非线性代数算子
- 主要用于二值图像，可扩展到灰度图像

https://en.wikipedia.org/wiki/Mathematical_morphology

数学形态学

- 数学形态学中，集合表示图像中的对象。
  - 在二值化图像中，所有白色像素的集合是该图像的完整形态学描述。
  - 问题中的集合是二维整数空间Z^2的元素，在此空间中，集合的每一个元素都是一个二维向量
- 形态学中的操作是基于结构元（SE）进行的，
- 结构元就是我们研究一幅图像中感兴趣特性所用的最小集合或子图象
- 一般结构元的原点都在对称中心处

数学形态学MM 发展历史

- **60**年代
  - 1964年，法国巴黎矿业学院，G.Matheron, J.Serra；铁矿的岩石断面定量分析，以预测其开采价值；
  - 1966年，南锡的酒吧， G.Matheron, J.Serra和Ph. Formeny奠定了数学形态学；
  - 1968年4月，法国成立枫丹白露(Fontainebleau)数学形态学研究中心；
- **70**年代
  - 以二值图像为主；TAS（纹理分析系统）；大量专利;
- **mid-70s to mid-80s**
  - 推广到灰度图
- **80-90**年代
  - 应用普及，得到广泛承认

目前，许多有效的图像处理系统是基于数学形态学方法原理设计的，有的把数学形态学算法纳入其基本软件(**ImageJ**)，并以其运算速度作为系统性能的重要标志之一

http://www.cmm.mines-paristech.fr/~serra/pdf/birth_of_mm.pdf

主要内容

- **预备知识：集合概念回顾**
- 膨胀与腐蚀(两个基本运算)
- 开运算与闭运算
- 击中击不中变换
- 应用举例

集合概念回顾

- A为Z^2中的一个集合， Z：整数集
- a = (a1, a2) 为A中的一个元素 $\boldsymbol{a \in A}$
- a 不是A的元素 $\boldsymbol{a \notin A}$
- 空集合: ∅

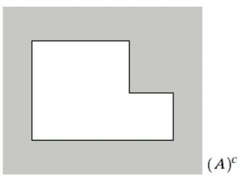
集合概念回顾

- 子集: $\boldsymbol{A \subseteq B}$ $\boldsymbol{A \subset B}$
- 并运算 $\boldsymbol{C = A \cup B}$
- 交运算 $\boldsymbol{C = A \cap B}$
- 不相交 $\boldsymbol{A \cap B = \emptyset}$

集合概念回顾

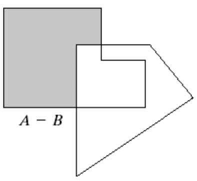
补集

$$A^c = \{w | w \notin A\}$$



差

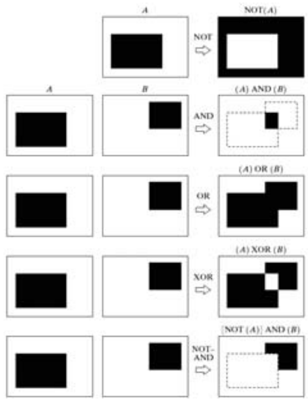
$$A - B = \{w | w \in A, w \notin B\}$$



二值图像的逻辑运算

- AND, OR, NOT
  - p, q为二值图像中的两个像素

p	q	p AND q (also $p \cdot q$)	p OR q (also $p + q$)	NOT (p) (also $\bar{p}$)
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0



前景：1：黑
背景：0：白

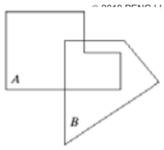
$A \cap B$

$A \cup B$

异或运算：不相交的，为真

$B - A$

形态学中的特定集合操作

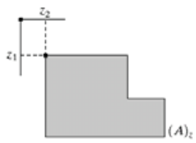


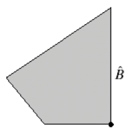
平移集
(移位Translation)

$(A)_z = \{c | c = a + z, \text{ for } a \in A\}$

对称集
(反射Reflection)

$\hat{B} = \{w | w = -b, \text{ for } a \in B\}$



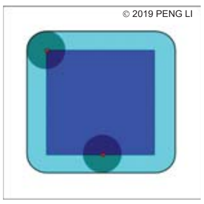


主要内容

- 预备知识：集合概念回顾
- **膨胀与腐蚀(两个基本运算)**
- 开运算与闭运算
- 击中击不中变换
- 应用举例

膨胀(Dilation)

胖了SE的一半

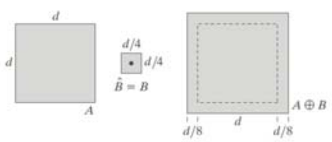


- 定义：

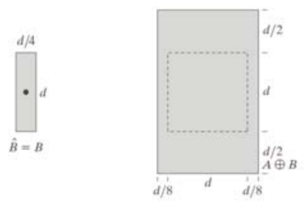
$$A \oplus B = \left\{ z \middle| (\hat{B})_z \cap A \neq \phi \right\}$$

  - A: 二值图像
  - B: 二值模板，称为结构元（structure element）
- 结构元B关于它的原点的映像$\hat{B}$进行z平移时，若得到的结果与集合A仍相交的部分，即为B对A膨胀的结果。
- 意义：
  - 膨胀是将与物体“接触”的所有背景点合并到该物体中，
  - 使边界向外部扩张的过程。
  - 可以用来填补物体中的空洞。（其中“接触”的含义由结构元描述）

膨胀运算示例

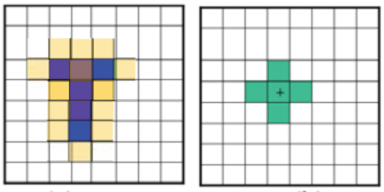


方形结构元将集合A的外围扩大



条形结构元将集合A上下边缘拓展

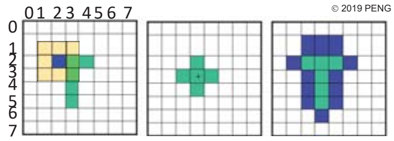
膨胀运算示例



- (a) (b)

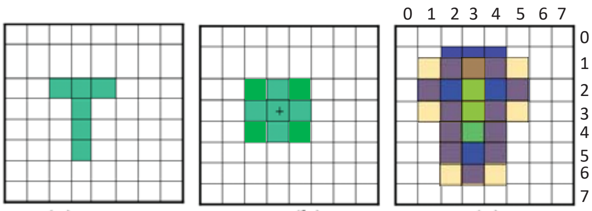
A *B* *A⊕B*

膨胀运算示例



- 对于上图(a)的图像以左上角位置为(0,0)，
- 结构元素以“＋”位置为参考点(0,0)。
- 则A和B分别表示为：
- A={(2,2),(2,3),(2,4),(3,3),(4,3),(5,3)}
- B={(0,0),(-1,0),(1,0),(0,-1),(0,1)}

- 用向量运算进行膨胀得到：
  $A \oplus B$=
  {(2,2),(2,3),(2,4),(3,3),(4,3),(5,3),
  (1,2),(1,3),(1,4),(2,3),(3,3),(4,3),
  (3,2),(3,3),(3,4),(4,3),(5,3),(6,3),
  (2,1),(2,2),(2,3),(3,2),(4,2),(5,2),
  (2,3),(2,4),(2,5),(3,4),(4,4),(5,4)}

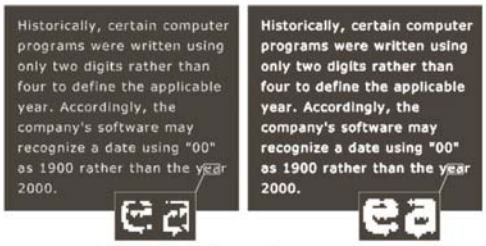


- (a) (b) (c)

膨胀：Bridging gaps in images

- 效果: 增加大小, 填补缺口

缺口大小由SE控制

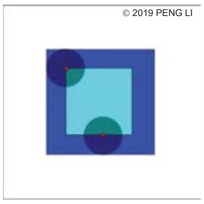


结构元

0	1	0
1	1	1
0	1	0

腐蚀(Erosion)

瘦了SE的一半



- 定义：

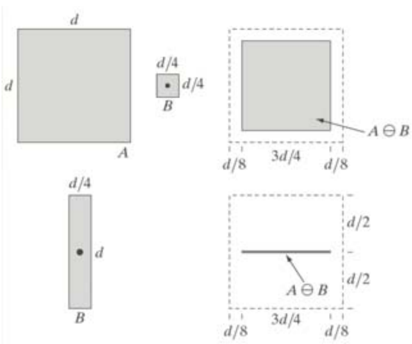
$$A \ominus B = \{z | (B)_z \subseteq A\}$$

  - A: 二值图像
  - B: 二值模板，称为结构元（structure element）

- 就是结构元B在对集合A中的每一个元素进行操作的结果中，仍然包含于A的全部元素的集合，即为B对A腐蚀的结果。

- 意义：
  - 腐蚀是一种消除边界点，使边界向内部收缩的过程。
  - 可以用来消除小且无意义的物体。

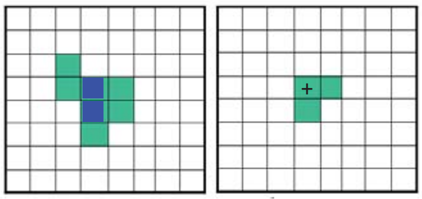
腐蚀运算示例



方形结构元将集合A的外围腐蚀掉

条形结构元将集合A腐蚀为一条线

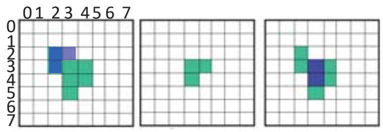
腐蚀运算示例



A　　B　　$A \ominus B$

可见，不能容纳结构元素的部分被腐蚀掉了。

腐蚀运算示例

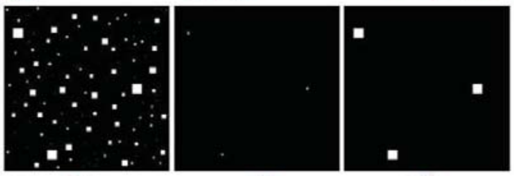


- 对于上图(a)的图像以左上角位置为（0，0），
- 结构元素以“＋”位置为参考点（0，0）。
- 则X和B分别表示为：
- A={(2,2),(2,3),(3,3),(4,3),(3,4),(4,4),(3,5)} 记为X
- B={(0,0),(1,0),(0,1)}

	(2,2)	(2,3)	(3,3)	(4,3)	(3,4)	(4,4)	(3,5)
b(0,0)	(2,2)	(2,3)	(3,3)	(4,3)	(3,4)	(4,4)	(3,5)
b(1,0)	(3,2)	(3,3)	(4,3)	(5,3)	(4,4)	(5,4)	(4,5)
b(0,1)	(2,3)	(2,4)	(3,4)	(4,4)	(3,5)	(4,5)	(3,6)
$\subseteq X$?							

✓ 使用腐蚀消除图像的细节部分，产生滤波器的作用

包含边长为1,3,5,7,9和15像素正方形的二值图像　　使用13×13像素大小的结构元素腐蚀原图像的结果　　使用13×13像素大小的结构元素膨胀图b，恢复原来15×15尺寸的正方形



a b c　图a　　图b　　图c

腐蚀操作去除某些图像成分

Using erosion to remove image components

486*486 binary image of a wire bond mask

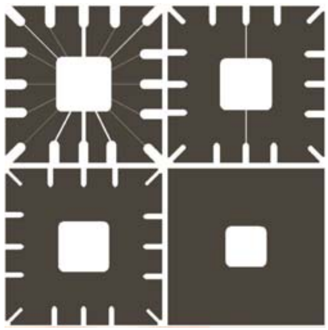


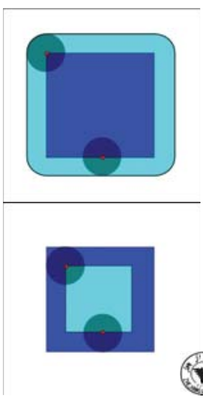
Image eroded using square structuring elements of size 11*11

Size 15*15

Size 45*45

膨胀与腐蚀：总结

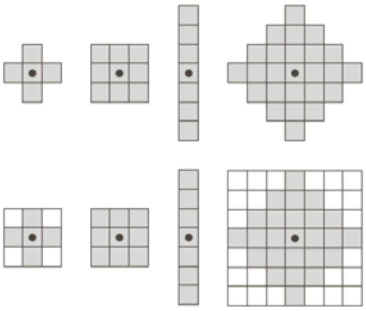
- 膨胀
  - 由B对A膨胀所产生的二值图象D是满足以下条件的点(x, y)的集合：
  - 如果B的原点平移到点(x, y)，那么它与A的交集非空
  - 即：有重叠，就算
- 腐蚀
  - 由B对A腐蚀所产生的二值图象E是满足以下条件的点(x, y)的集合：
  - 如果B的原点平移到点(x, y)，那么B将完全包含于A中



膨胀与腐蚀: 对偶关系

- 膨胀与腐蚀彼此关于集合求补运算和反射运算是对偶的,即：
  - $(A \ominus B)^c = A^c \oplus \hat{\boldsymbol{B}}$　B对A的腐蚀是$\hat{\boldsymbol{B}}$对A^c膨胀的补集　（1）
  - $(A \oplus B)^c = A^c \ominus \hat{\boldsymbol{B}}$　B对A的膨胀是$\hat{\boldsymbol{B}}$对A^c腐蚀的补集　（2）
- 证明：
  - 由腐蚀的定义：$(A \ominus B)^c = \{z \mid (B)_z \subseteq A\}^c$
  - 若集合$(B)_z$包含于集合A，则$(B)_z \cap A^c = \phi$
  - 所以$(A \ominus B)^c = \{z \mid (B)_z \cap A^c = \phi\}^c$
  - 由膨胀的定义$A \oplus B = \left\{z \middle| (\hat{\boldsymbol{B}})_z \cap A \neq \phi\right\}$，得到
  - $A^c \oplus \hat{\boldsymbol{B}} = \{z \mid (B)_z \cap A^c \neq \phi\}$
  - 故：$(A \ominus B)^c = \{z \mid (B)_z \cap A^c = \phi\}^c = A^c \oplus \hat{\boldsymbol{B}}$　（1）成立，（2）同理。

结构元

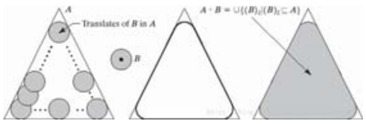


主要内容

- 预备知识：集合概念回顾
- 膨胀与腐蚀(两个基本运算)
- 开运算与闭运算
- 击中击不中变换
- 应用举例

开运算

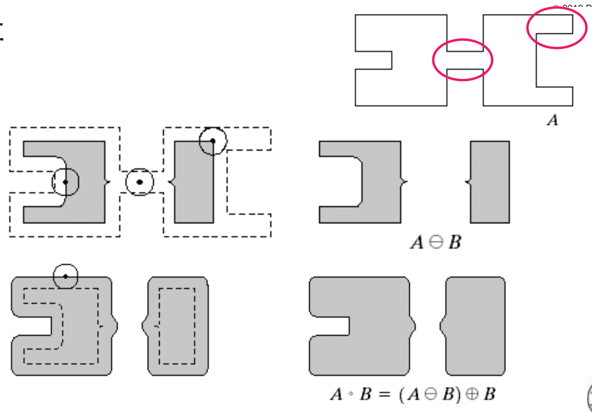
- 定义：$A \circ B = (A \ominus B) \oplus B$
  - 先腐蚀，后膨胀



- Opening= erosion + dilation
- Erosion + dilation = original image ?

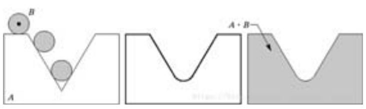
- 用来消除小物体、
- 在纤细点处分离物体、
- 平滑较大物体的边界的同时并不明显改变其面积。

开运算



闭运算

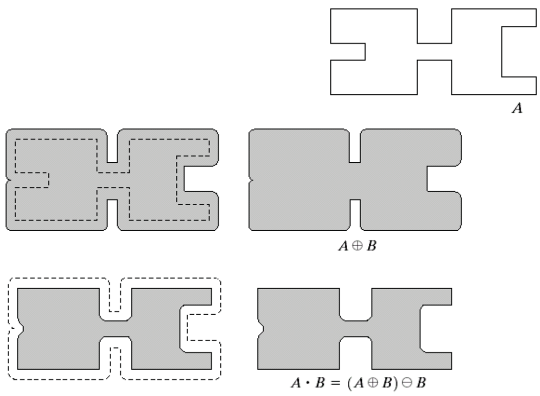
- 定义：$A \cdot B = (A \oplus B) \ominus B$
  - 先膨胀，后腐蚀



- Closing = dilation + erosion
- Dilation+erosion = erosion + dilation ?

- 用来填充物体内细小空洞、
- 连接邻近物体、
- 平滑其边界的同时并不明显改变其面积。

闭运算



性质

- 开运算
  I. $A \circ B$ is a subset (subimage) of A
  II. If $C \subseteq D$, then $C \circ B \subseteq D \circ B$
  III. $(A \circ B) \circ B = A \circ B$

open后变小

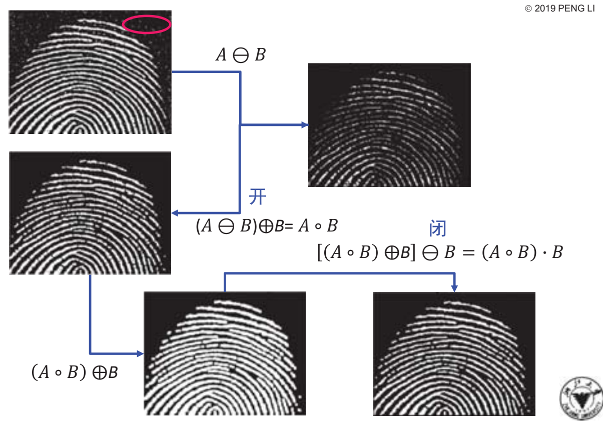
多次open等于一次open

- 闭运算
  I. A is a subset (subimage) of $A \cdot B$
  II. If $C \subseteq D$, then $C \cdot B \subseteq D \cdot B$
  III. $(A \cdot B) \cdot B = A \cdot B$

close后变大

多次close等于一次close

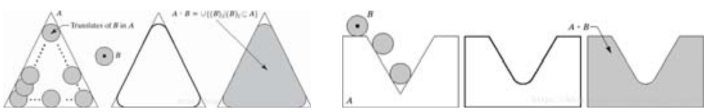
噪声图像



开、闭运算的意义

- 开运算：通常对图像轮廓进行平滑，使狭窄的“地峡”形状断开，去掉细的突起。

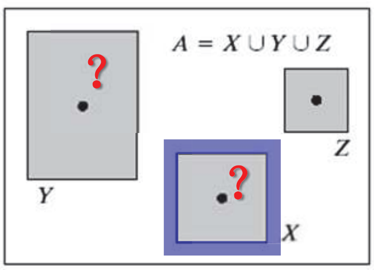
- 闭运算：也是趋向于平滑图像的轮廓，但于开运算相反，它一般使窄的断开部位和细长的沟熔合，填补轮廓上的间隙。



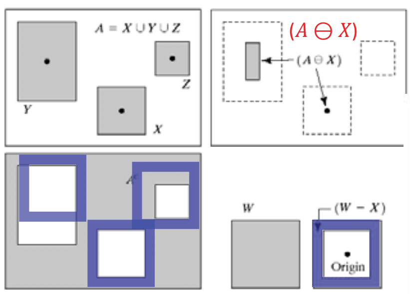
主要内容

- 预备知识：集合概念回顾
- 膨胀与腐蚀(两个基本运算)
- 开运算与闭运算
- 击中击不中变换
- 应用举例

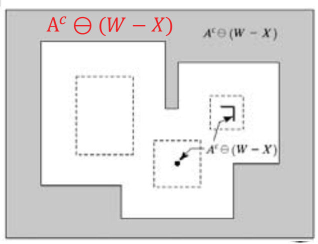
击中和击不中变换（Hit-or-miss transformation，HMT ）



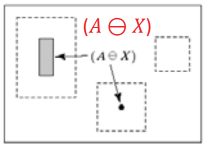
HMT: Detect object via background

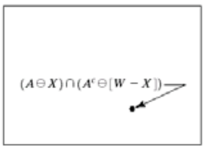


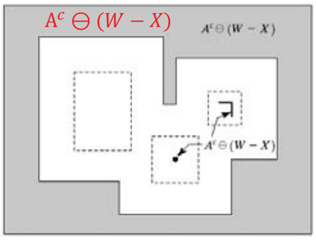
$A \circledast B = (A \ominus X) \cap [A^c \ominus (W - X)]$



HMT: Eliminate un-necessary parts







$A \circledast B = (A \ominus X) \cap [A^c \ominus (W - X)]$

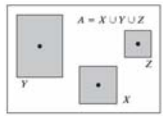
HMT

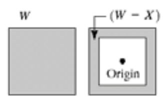
- HMT定义为:

$$A \circledast B = (A \ominus B_1) \cap [A^c \ominus B_2]$$

$$A \circledast B = (A \ominus X) \cap [A^c \ominus (W - X)]$$

  – 二值模板(结构元) B由两部分组成(B1,B2),
  – B1物体点，B2背景点
  – B称为混合结构元



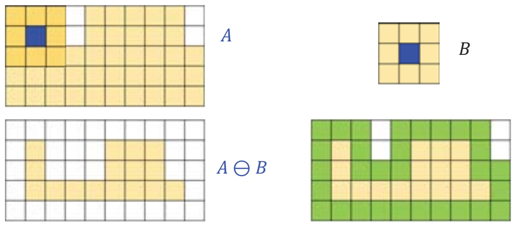


- HMT是形态学运算推广到更为一般的情况,
- 这时结构元不仅含有**物体点**，而且还含有**背景点**,
- 只有当结构元素与所对应的区域完全符合时才作为结果输出到输出图象。
- 实际上就演变为条件**严格的模板匹配**。
- 一个基本的形状检测工具

主要内容

- 预备知识：集合概念回顾
- 膨胀与腐蚀(两个基本运算)
- 开运算与闭运算
- 击中击不中变换
- **应用举例**

边界提取

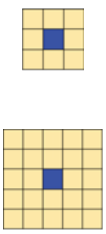


- 边界提取定义：

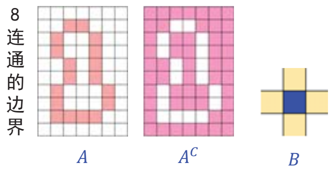
$$\beta(A) = A - (A \ominus B)$$

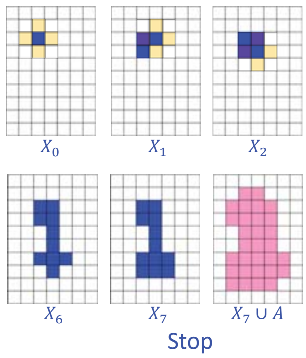
- 即：先用B对A进行腐蚀，然后用A减去腐蚀得到的结果
- 这里结构元B是一个自定义的合理结果，取决于我们希望的边缘的厚度。

边界提取



孔洞填充





- 孔洞填充定义：

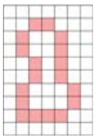
  $X_k = (X_{k-1} \oplus B) \cap A^c$, k=1,2,3…
  – B为一个对称结构元
  – X_0为包含于孔洞的一个点
  – 逐次迭代，直到X_k=X_{k-1}，结束迭代。

孔洞填充

- 孔洞填充的目的：当已知孔洞位置后，应用形态学的方法填充所有孔洞。
- 孔洞定义为：由前景像素相连的边界所包围的一个背景区域。
- 孔洞填充定义：

  $X_k = (X_{k-1} \oplus B) \cap A^c$, k=1,2,3…

  – B为一个对称结构元
  – X_0为包含于孔洞的一个点
  – 逐次迭代，直到X_k=X_{k-1}，结束迭代。



- 条件膨胀：
  – 利用A^c的交集将结果限制在感兴趣的区域内
  – 如果对上述公式的左部不加限制，则上述膨胀将填充整个区域。

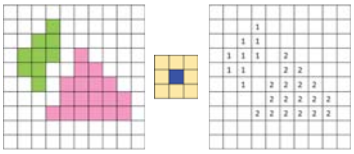
区域填充——例子

Original image | The first filled region | Fill all regions

提取连通分量

- **算法：**
- 初始化：X_0=连通分量A_1中的某个点
- 循环：*Do* $X_k = (X_{k-1} \oplus B) \cap A$

  Until X_k=X_{k-1}

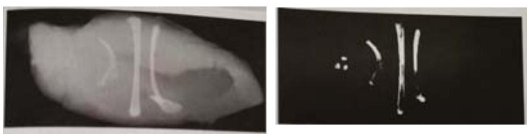


- 提取连通分量也是标注连通分量的过程，给每个连通区域分配一个唯一代表该区域的编号
- 与孔洞填充相反
  – 提取连通分量是根据选定的点找到图像中包含该点的白色区域
  – 孔洞填充是找到包含选定的点的黑色区域

提取连通分量的应用

- 计算某一连通分量的大小
  – 计算该编号对应的像素数目
- 计算某一连通分量的质心
  – 计算该编号对应像素坐标的平均值

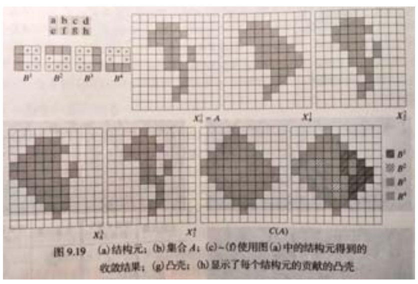
提取连通分量——检测包装食品中的外来物



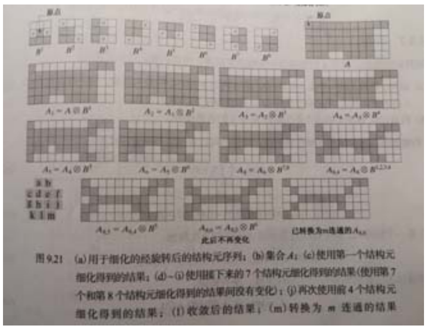
连通分量	连通分量中的像素数	连通分量	连通分量中的像素数
01	11	09	7
02	9	10	11
03	9	11	11
04	39	12	9
05	133	13	9
06	1	14	674
07	1	15	85
08	743		

凸壳

- 凸壳：如果在集合A内任意链接两个点的直线都在A的内部，则集合A是凸形的。



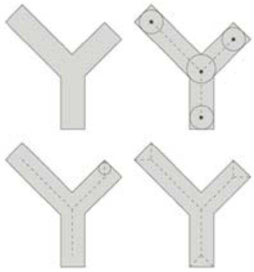
细化



M连通的细化集合，
消除多重路径

骨架

- 集合A中的骨架S(A)可由下图做出直观解释。
- S(A)是包含于集合A且与集合A两个以上边界相切的全部圆的圆心的集合。

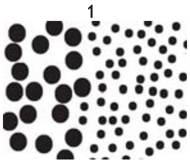


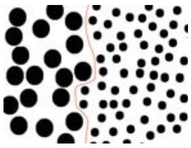
小结

- 预备知识：集合概念回顾
- 膨胀与腐蚀(两个基本运算)
- 开运算与闭运算
- 击中击不中变换
- 应用举例
  - 边界提取
  - 孔洞填充
  - 连通分量提取
  - 凸壳
  - 细化
  - 骨架

讨论

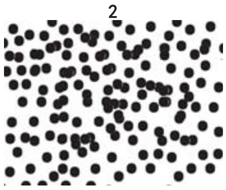
1





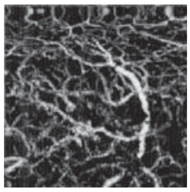
预期效果：找出下图中这条边界

2



假设：颗粒大小一致，设计算法，区分：
1）在边界不不完整的颗粒
2）彼此重叠的颗粒
3）完整独立立的颗粒

3



形态学参数：
血管面积密度
血管骨架密度
血管直径
血管周长
等