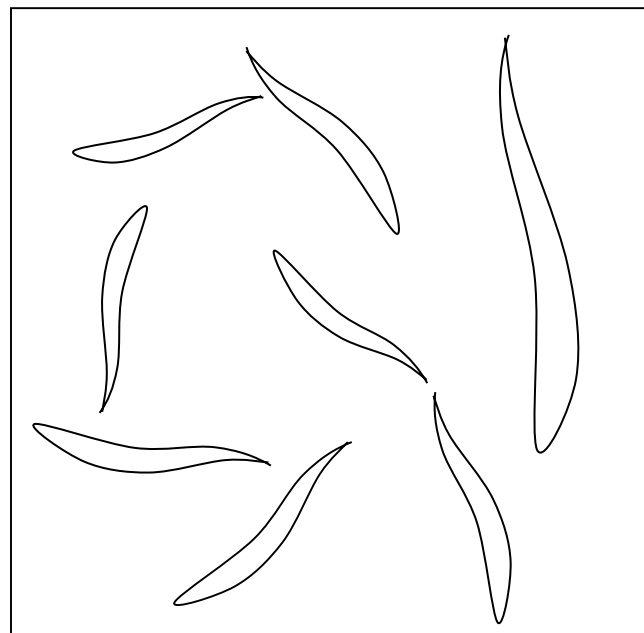


二值图像分析

Binary Image Analysis

一、二值图像

二值图像与线图形是描述对象物形状的较为简单的形式，而线图形则更为简洁、直观。



(1) 二值图象的特点

- a) 只有两个灰度级:1 (255) 和0;
- b) 图象中的特征如边缘、轮廓可用二值图像表示;
- c) 二值图像处理算法简单, 所需的内存小, 计算速度快;
- d) 二值视觉系统技术可用于灰度图像视觉系统。

(2)二值图象的获取

a) 硬件实现

敏感元二值输出或逻辑输出。敏感元模拟值输出，通过硬件电路二值化。

b) 软件实现

灰度图象可以通过**阈值 (threshold) 分割**进行二值化处理。

(3)灰度图象的二值化

设一幅灰度图像中物体的灰度分布在某一区间内，经过阈值运算后的图像为二值图像。

$$g[i, j] = \begin{cases} 255 & \text{如果 } T_1 \leq f[i, j] \leq T_2 \\ 0 & \text{其它} \end{cases}$$

$$g[i, j] = \begin{cases} 255 & \text{如果 } f[i, j] \in Z \\ 0 & \text{其它} \end{cases}$$

其中Z是组成物体各部分灰度值的集合。

二、几何特性

通过阈值化方法检测出物体后，下一步就要对物体进行识别和定位。

在多数工业应用中，摄像机的位置和环境是已知的，因此通过简单的几何知识就可以从物体的二维图像确定出物体的三维位置。

利用尺度和形状特征来识别：如大小、位置和方向。

2.1 尺寸和位置

一幅二值图像区域的面积

$$A = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} f[i, j]$$

物体的形心位置为：

$$\left\{ \begin{array}{l} \bar{x} = \frac{\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} jf[i, j]}{A} \\ \bar{y} = -\frac{\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} if[i, j]}{A} \end{array} \right.$$

其中 \bar{x} 和 \bar{y} 是区域相对于左上角图像的中心坐标。

2.2 密集度和体态比

区域的密集度（Compact）可用下面的式子来度量：

$$C = \frac{A}{p^2}$$

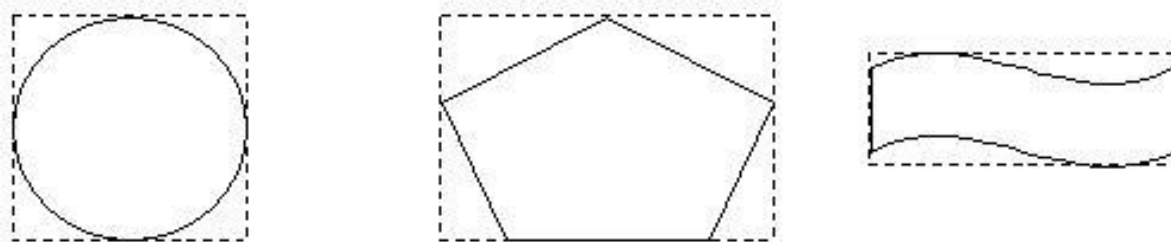
其中， p 和 A 分别为图形的周长和面积。根据这一衡量标准，圆是最密集的图形，其密集密度为最大值 $\frac{1}{4\pi}$ ，其它一些图形的比值要小一些。

正方形的密集度=？

对于数字图像， A/p^2 是指物体像素点数量除以其边界长度的平方。这是一种很好的散布性或密集性度量方法。这一比值在许多应用中被用作为一个区域的一个特征。

2.2 密集度和体态比

体态比定义为区域的最小外接矩形的长与宽之比，正方形和圆的体态比等于1，细长形物体的体态比大于1。下图所示的是几种形状的外接矩形。

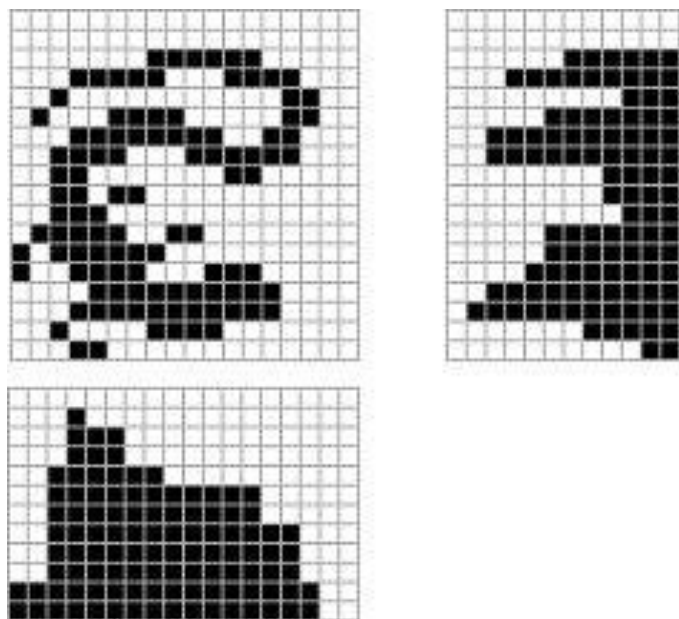


几种外接矩形示意图

2.3 投影

给定一条直线，用垂直该直线的一簇等间距直线将一幅二值图像分割成若干条，每一条内像素值为1的像素个数为该条**二值图像在给定直线上的投影**（projection）。

当给定直线为水平或垂直直线时，计算二值图像每一列或每一行上像素值为1（255）的像素数量，就得到了二值图像的水平 and 垂直投影。



（1）由于投影包含了图像的许多信息，所以投影是二值图像的一种简洁表示方式。

（2）显然，投影不是唯一的，同样的投影可能对应不同的图像。

一幅二值图像及其水平和垂直投影图

2.3 投影

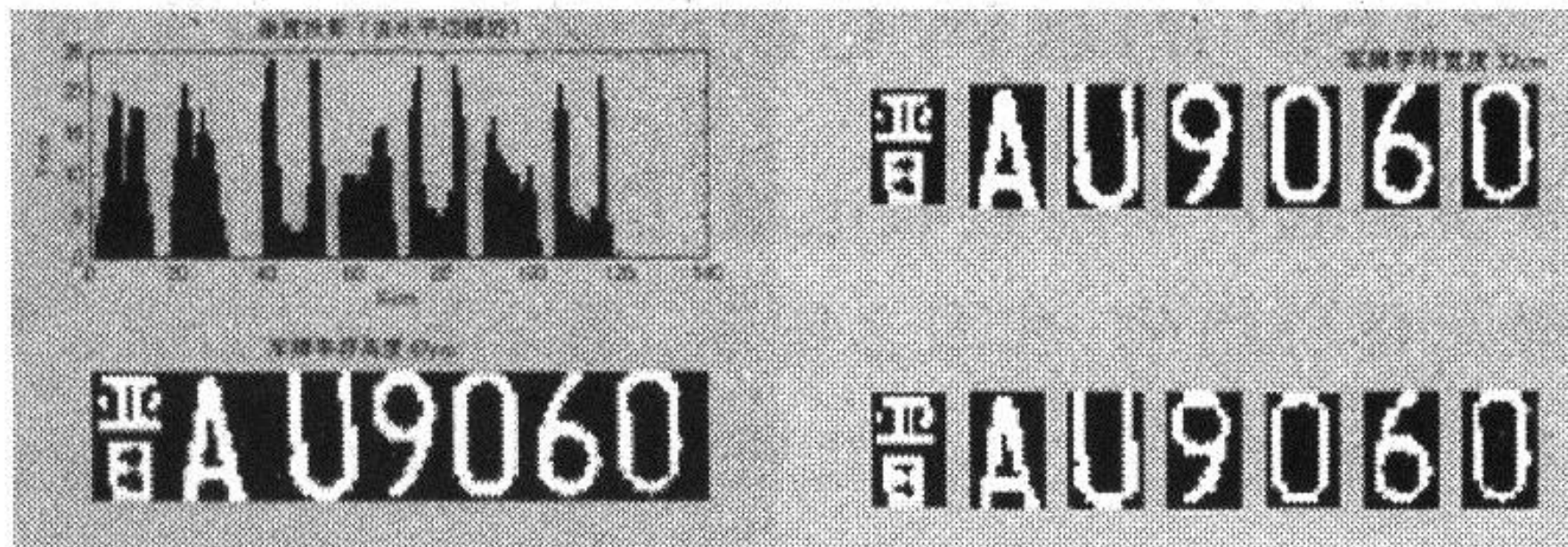
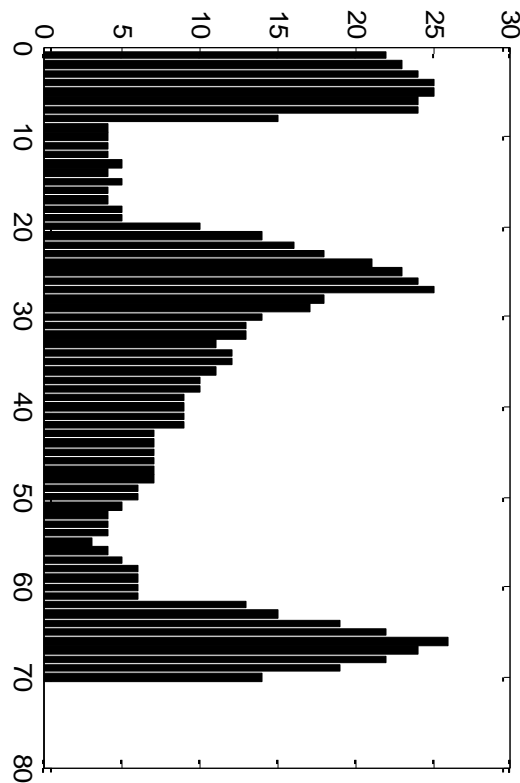
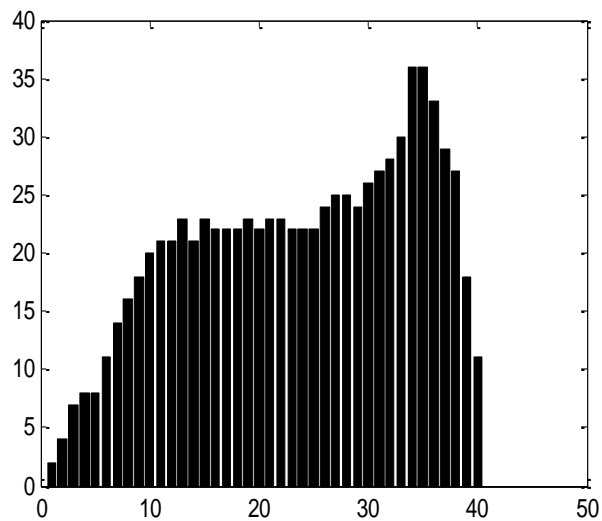


图 4 车牌字符高度、宽度及分割的字符

2.3 投影



演示

2.3 投影

计算文字的行数？

是一部具有世界影响力的人情小说作品，举世公认的中国古典小说以贾、史、王、薛四大家族的兴衰为背景，以贾府的家庭琐事为主线，描写了以贾宝玉和金陵十二钗为中心的正邪两赋有的悲剧命运预见封建社会必然走向灭亡，揭示出封建末世危机

三、二值图像的分割



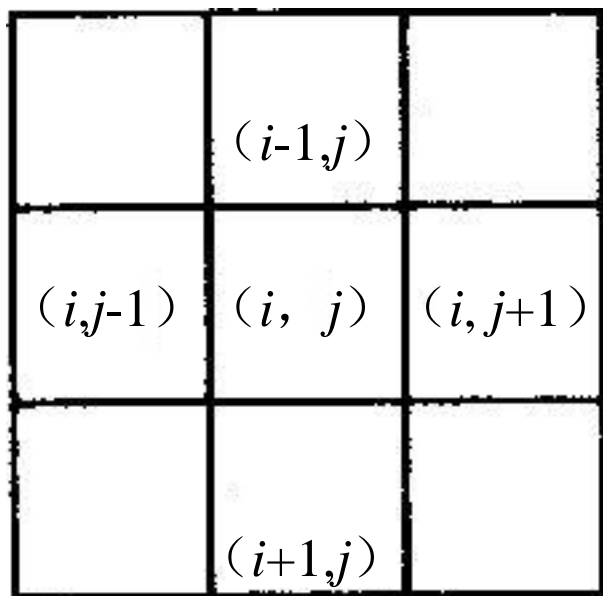
二值图像的分割处理：对同灰度值的像素，分别加上标记，从而来描述对象物的形状。

1、连接性：

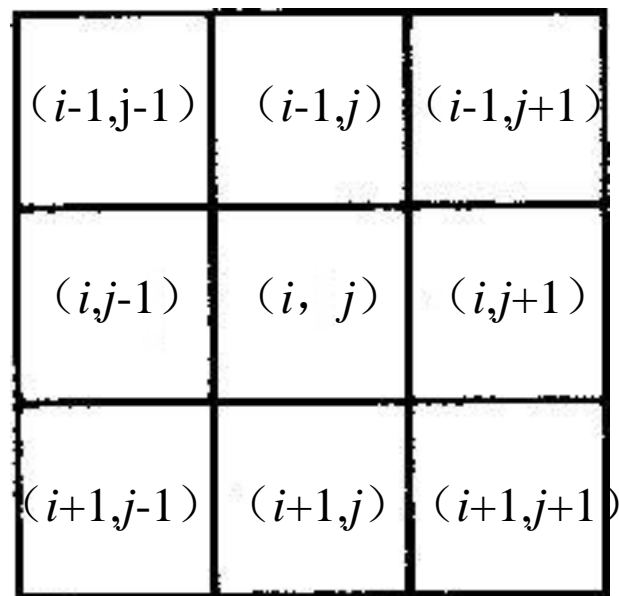
- 二值图像中像素之间的关系常用连接性表示；
- 连接性包含：连接与非连接；

4连通指的是从区域上一点出发，可通过4个方向，即上、下、左、右移动的组合，在不越出区域的前提下，到达区域内的任意像素；8连通方法指的是从区域上一点出发，可通过左、右、上、下、左上、右上、左下、右下这8个方向的移动组合来到达区域内的任意像素。

a、4邻域或4邻点；

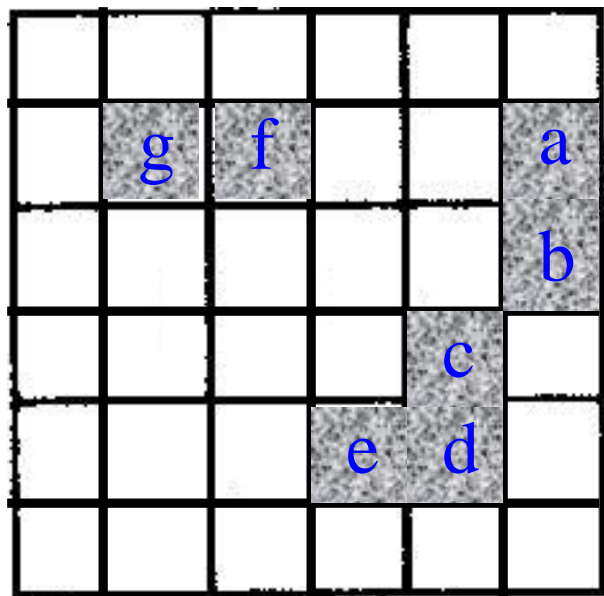


b、8邻域或8邻点；



■ **连接性：** 在一组像素中，如果所有像素值都相同（0或1），并依次互为邻点，那么这组像素中的任意两个像素间都认为存在连接关系的，否则，两者为非连接关系。

采用不同的邻域，像素间连接关系也不同。如：



■ 为1-像素，用来表示对象物

□ 为0-像素，用来表示背景

• 8邻域连接：

a-b-c-d-e ;

f-g 。

• 4邻域连接：

a-b ; c-d-e ;

f-g 。

2、连接成分：

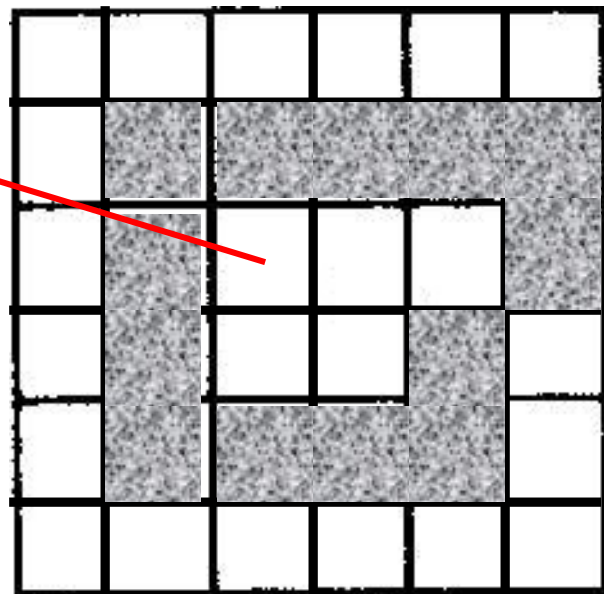
二值图像中有连接关系的像素的集合称为连接成分，也就是区域

- 1-像素连接成分：指对象，也称图形分量；
- 0-像素连接成分：指背景或孔。（孔指的是与图像周边不相连，或者说是被对象物所包围的**0-像素**的集合）

3、加标记：

- 对已分割的各个图形分量（1-像素连接成分），应加上不同的标记加以区分；

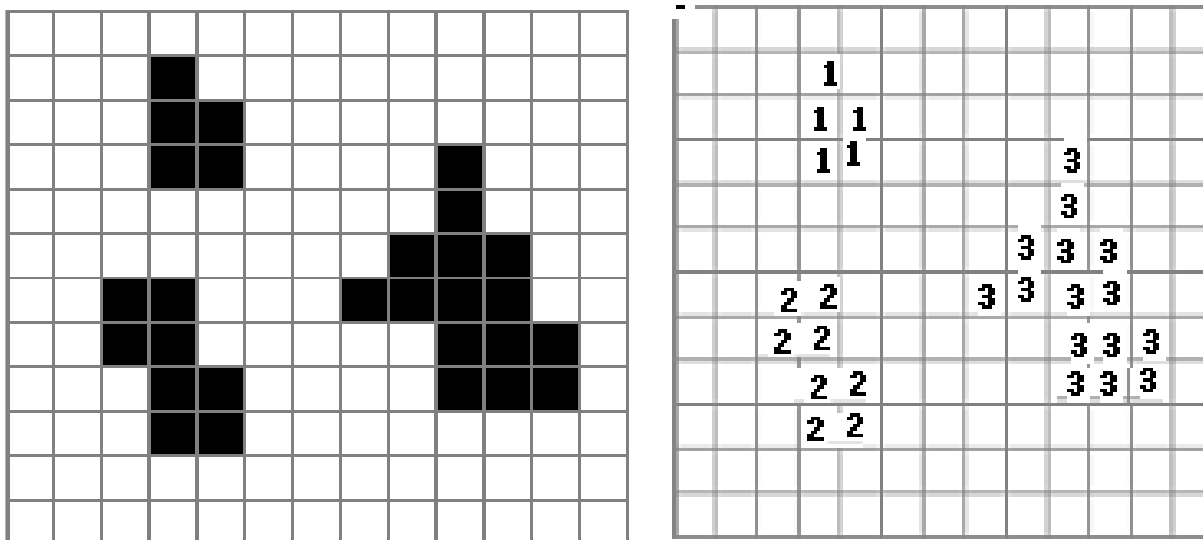
孔



3.1 连通成分标记算法

图像识别中常见的运算是找连通成分。

连通标记算法可以找到图像中的所有连通成份，并对同一连通成份中的所有点分配同一标记。下图表示的是一幅图像和已标记的连通成份。下面介绍两种连通成份标记算法：**递归算法和序贯算法**



算法1：连通成份递归算法

1. 扫描图像，找到无标记且值为1的像素点，给它分配一个标记 L ;
2. 递归分配标记 L 给1点的邻点;
3. 如果不存在没标记的点，则停止;
4. 返回第1步。

算法2：4-连通成份序贯算法

	$(i-1, j)$	
$(i, j-1)$	(i, j)	$(i, j+1)$
	$(i+1, j)$	

1. 从左至右、从上到下扫描图像。
2. 如果当前像素点灰度为1，则：
 - (a) 如果上面点和左面点只有一个标记，则复制这一标记；
 - (b) 如果这两个点有相同的标记，复制这一标记；
 - (c) 如果这两个点有不同的标记，则复制上点的标记且将两个标记输入等价表中作为等价标记；
 - (d) 否则给这一个像素点分配一新的标记并将这一标记输入等价表；
3. 回到第2步直到扫描完所有点。
4. 在等价表的每一等价集中找到最低的标记。
5. 扫描图像，用等价表中的最低标记取代每一标记。

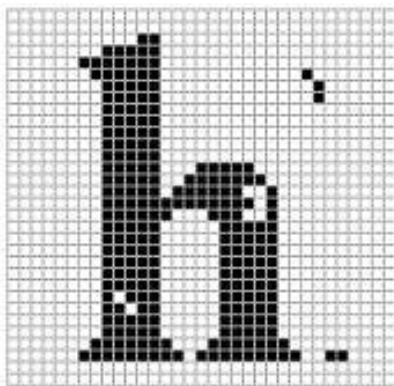
四、二值图像的平滑（收缩与扩张）

1、噪声的表现形式：

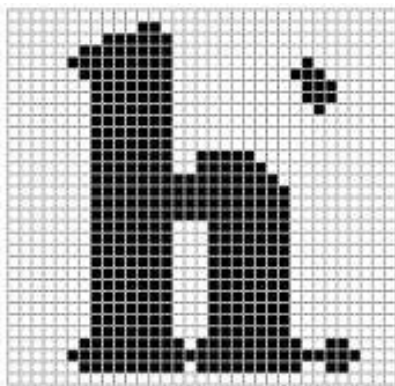
- 小孔：指相对面积较小的被对象物包围的0-像素连接成分；
- 点状图形：指相对面积较小的1-像素连接成分。

2、处理方法：

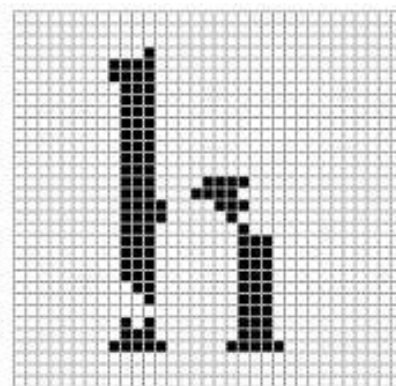
- **收缩处理（腐蚀）**：反复去除图形表面像素，将图形逐步缩小，以达到消去点状图形的效果；
- **扩张处理（膨胀）**：将图形表面不断扩散，以达到去除小孔的效果。



原始噪声图像



扩展运算



收缩运算

■ 收缩处理:

- a、**图形表面**指图像分量中与0-像素相邻的那些像素的集合;
- b、当某1-像素的邻点中存在0-像素, 则此1-像素被称为图像表面像素;
- c、去除图形表面像素的方法是将表面1-像素变为0-像素。
- d、采用4邻域或8邻域时的处理算子:

$$g(i, j) = \begin{cases} 0 & f(i, j) = 0 \\ 0 & f(i, j) = 1 \text{ 且其邻域中存在一邻点为 } 0 \\ 1 & \text{其他} \end{cases}$$

■ 扩张处理:

- a 含义: 将与图形表面相邻的0-像素变1-像素。
- b 采用4邻域或8邻域时的处理算子:

$$g(i, j) = \begin{cases} 1 & f(i, j) = 1 \\ 1 & f(i, j) = 0 \text{ 且其邻域中存在一邻点为1} \\ 0 & \text{其他} \end{cases}$$

注: 由于收缩 (能消去点状图形, 但会扩大小孔) 与扩张 (能消去小孔, 但会扩大点状图形) 得到是两种相反的处理结果, 因而在二值图像的平滑过程, 必须对两者进行适当的组合, 交叉使用。

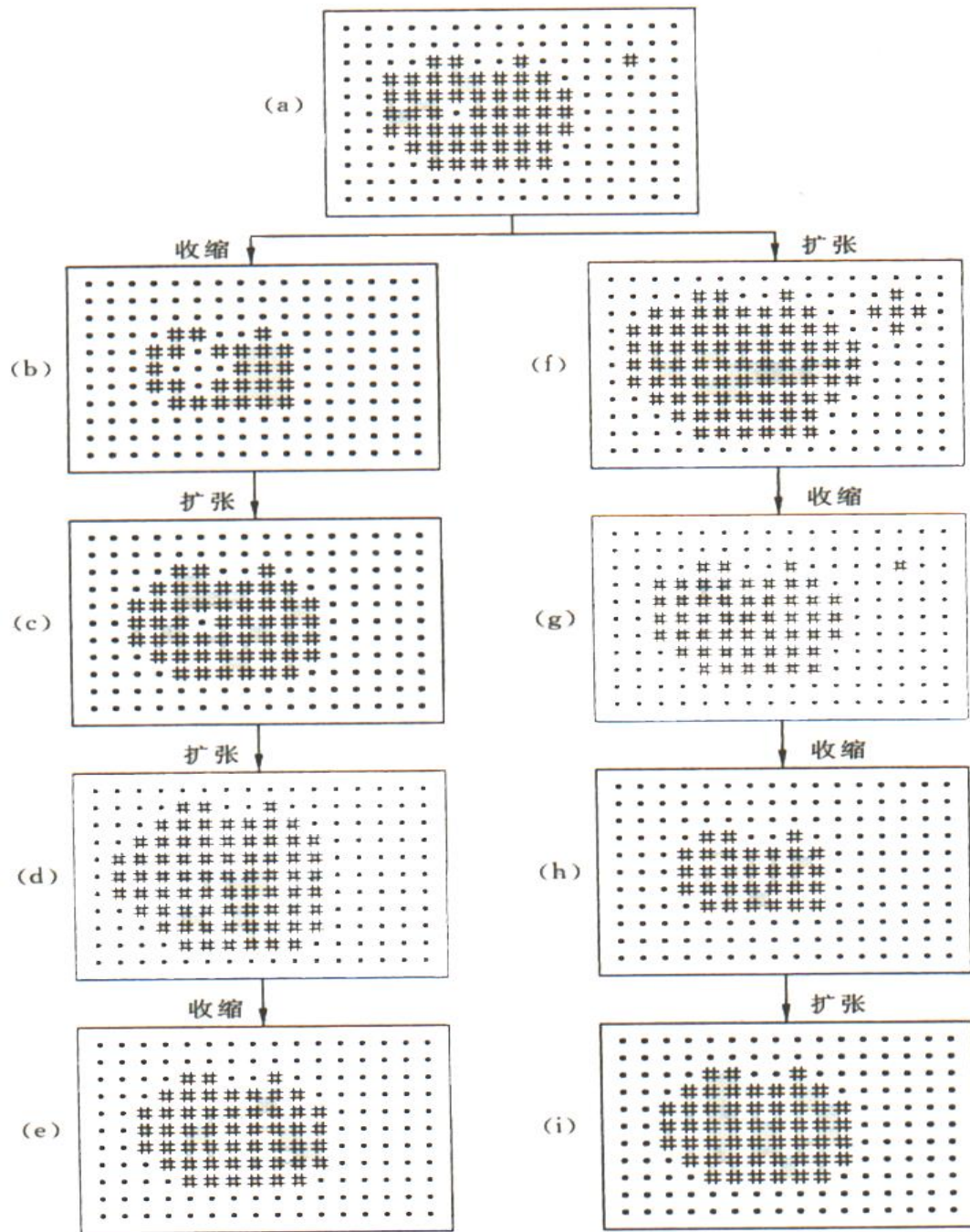


图 4.24 二值图像平滑处理(4 邻域)

3、细线化:

用细线条来构成的线图形来表示图像的方法;

- 优点: 存储量小, 便于识别;

- 基本要求:

- (1) 线宽为一个像素;

- (2) 细线位置基本处于原线宽的中心;

- (3) 保持图形的连接性不变, 并且不能出现孔和点的新出现或消失现象;

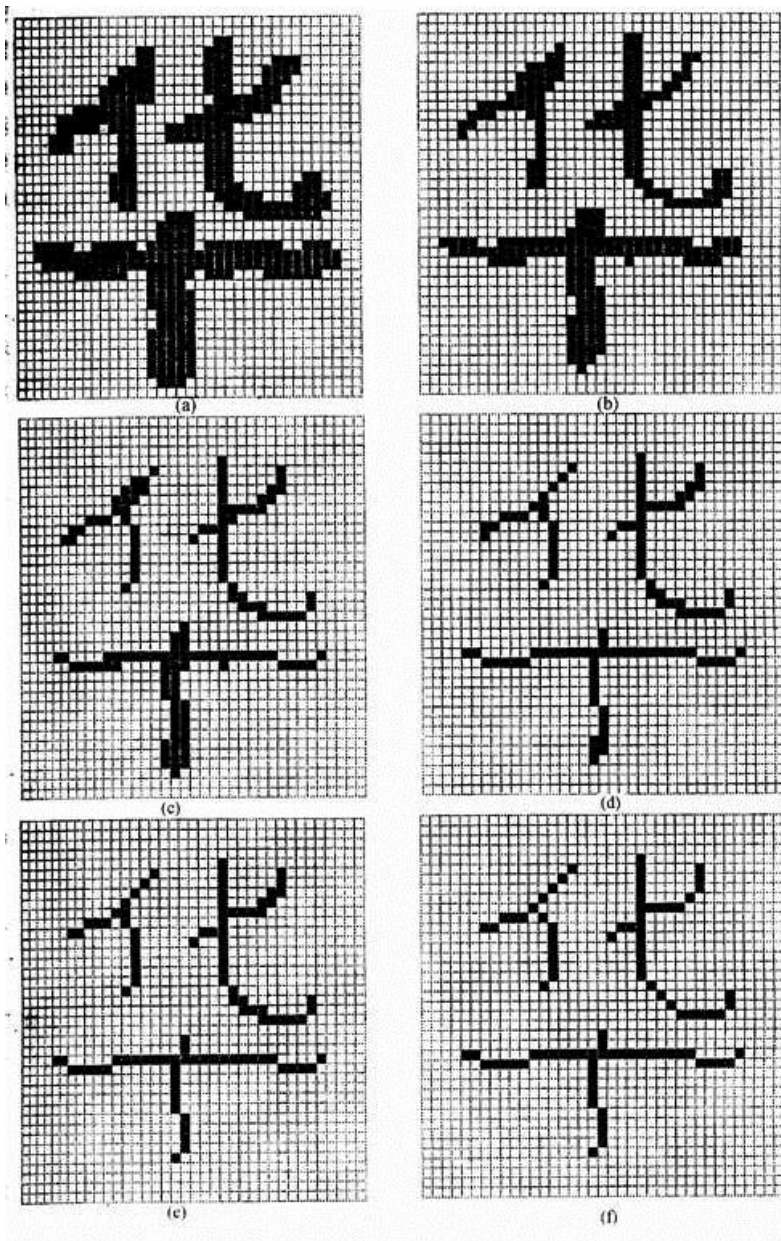
- (4) 图形端部基本不缩短。

- 细线化过程的实质: 在保持图像的连接性和图形长度不变的前提下求出图形的中心线



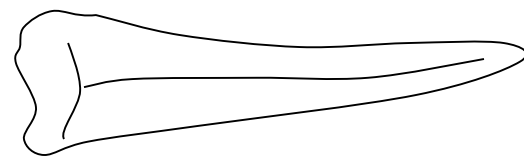
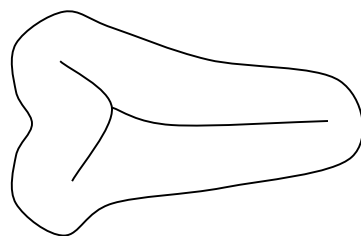
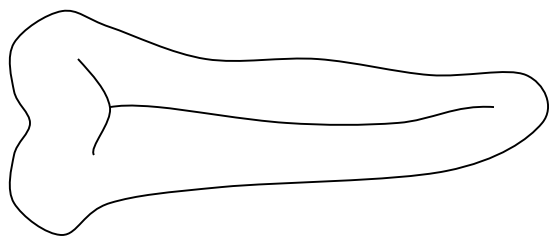
算法1: 4 邻点细化迭代算法

- 1、对于每一个像素，如果
 - 1) 没有上近邻（或下近邻或左近邻或右近邻）
 - 2) 不是孤立点或终止线
 - 3) 去除该像素点不会断开区域则去除该像素点.
- 2、重复步骤1直到没有像素点可以去除为止。

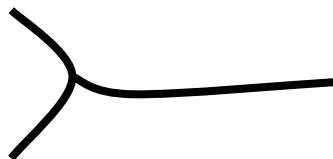


细化“华”的迭代过程：

(a) 原图像，(b)-(f)为五次迭代过程，每次迭代削去一层边界。



胚芽的轮廓



胚芽的细线化处理

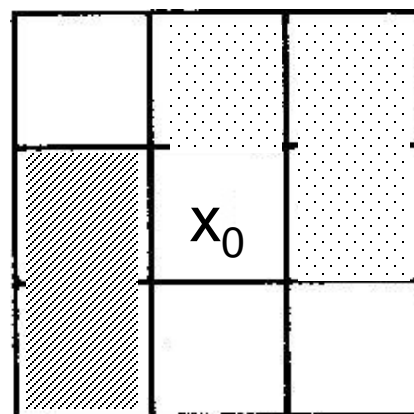
算法2:

■ 连接数 N_c :

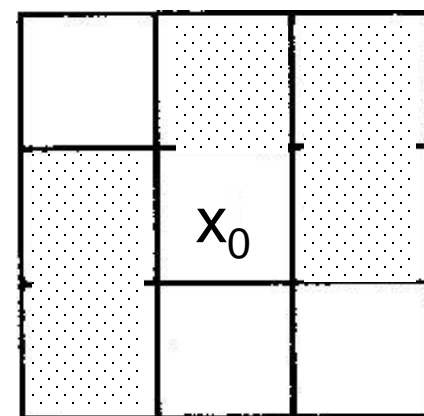
①、定义：在中心为1-像素 x_0 的 3×3 邻域中，与 x_0 有连接关系的图形分量(不包含 x_0)的个数。

x_4	x_3	x_2
x_5	x_0	x_1
x_6	x_7	x_8

	1	1
1	x_0	1
1		



4邻域连接



8邻域连接

4邻域连接时 $N_c = 2$;

8邻域连接时 $N_c = 1$;

②、连接数的计算： 采用4-连接时： $N_c^{(4)} = \sum_{k \in S} (x_k - x_k x_{k+1} x_{k+2})$
 采用8-连接时： $N_c^{(8)} = \sum_{k \in S} (\bar{x}_k - \bar{x}_k \bar{x}_{k+1} \bar{x}_{k+2})$
 式中： $S=\{1, 3, 5, 7\}$ ，并规定 $x_9 = x_1$ ，其中， $\bar{x}_k = 1 - x_k$

- 保持连接性条件： $N_C=1$
- 去除1-像素条件：细线化过程即去除1-像素过程
 - ① 去除1-像素必须是表面1-像素；
 - ② 满足保持连接性条件 $N_C = 1$ ；
 - ③ 去除1-像素不是线图形端点；

x_4	x_3	x_2
x_5	x_0	x_1
x_6	x_7	x_8

非图形端点的条件为中心1-像素 x_0 的邻域中0-像素的个数 $N < 7$



www.123rf.com

