

# PROJECT 09 -形态学处理

吴江南 3160104699

## 一、 实验内容介绍

- (1) 编写二值图像的腐蚀/膨胀函数；
- (2) 编写二值图像的交差补基本运算函数；
- (3) 对灰度图像重复上面两个步骤；
- (4) 利用(1)~(3)中编写的函数分别对二值图像和灰度图像进行腐蚀、膨胀、交差补、开合运算、边界提取、形态学梯度分析等等；
- (5) 实验主要涉及

### **PROJECT 09-01** [Multiple Uses] Morphological and Other Set Operations

(a) Write a computer program capable of performing binary dilation and erosion with an arbitrary structuring element of size  $3 \times 3$  that can be specified by the user.

(b) Write a computer program for performing set intersection, differencing, and complementation (See Section 2.6.4).

### **PROJECT 09-02** [Multiple Uses] Boundary Extraction

(a) Use your results from Project 09-01 to implement morphological boundary extraction as in Eq. (9.5-1).

(b) Download Fig. 9.14(a) from the book web site and extract its boundary.

## 二、 函数说明

子函数: DilateErode\_BinaryImg.m —— 二值图像的膨胀腐蚀

DilateErode\_GrayImg.m —— 灰度图的膨胀腐蚀

complement.m —— 灰度图的补操作

`differencing.m` ——灰度图的差操作

`intersection.m` ——灰度图的交操作

主函数: `mainBinary.m` ——二值图像的测试函数

`mainGray.m` ——灰度图的测试函数

### 三、 实验原理

#### (一) 二值图像的形态学和集合操作原理

1、二值图像腐蚀：对  $Z$  中的集合  $A$  和  $B$ ，使用  $B$  对  $A$  进行腐蚀定义为：

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

这个公式说明，使用结构元素  $B$  对  $A$  进行腐蚀是所有  $B$  中包含于  $A$  中的点  $z$  的集合用  $z$  平移。

2、二值图像膨胀：设  $A$  和  $B$  是  $(Z^2)$  中的集合， $A$  被  $B$  膨胀定义为：

$$A \oplus B = \{z | (\hat{B})_z \cap A \neq \emptyset\}$$

这个公式是以得到  $B$  的相对于它自身原点的映像并且由  $z$  对映像进行位移为基础的。 $A$  被占膨胀是所有位移  $z$  的集合，这样， $B$  和  $A$  至少有一个元素是重叠的。与在其他形态学运算中一样，集合  $B$  通常叫做膨胀的结构函数。

3、边界提取：由  $B$  对  $A$  腐蚀，而后用  $A$  减去腐蚀得到：

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

其中  $\beta(A)$  表示集合  $A$  的边界。

#### (一) 灰度图像的形态学和集合操作原理

1、膨胀：  $f \oplus B$

$$(f \oplus B)(s, t) = \max\{f(s - x, t - y) + b(x, y) | (s - x), (t - y) \in D_f, (x, y) \in D_b\}$$

这里  $D_f$  和  $D_b$  分别是  $f$  和  $b$  的定义域。 $f$  和  $b$  是函数而不是二值形态学情况中的集合。 $(s - x), (t - y)$  必须在  $f$  的定义域内以及  $x$  和  $y$  必须在  $b$  的定义域内的条件与膨胀的二值定义中的条件是相似的(这里两个集合的交集至少应有一个元素)。该公式的形式与二维卷积是相似的，并且用最大值运算代替卷积求和，用加法运算代替卷积乘积。

2、腐蚀：  $f \ominus B$

$$(f \ominus B)(s, t) = \min\{f(s + x, t + y) - b(x, y) | (s + x), (t + y) \in D_f, (x, y) \in D_b\}$$

这里  $D_f$  和  $D_b$  分别是  $f$  和  $b$  的定义域。平移参数  $(s + x), (t + y)$  必须在  $f$  的定义域内，而且  $x$  和  $y$  必须在  $b$  的定义域内，这与腐蚀的二值定义中的条件相似。同样，公式在形式上与二维相关是相似的，并且用最小值运算代替了相关运算，用减法运算代替了相关的乘积。

3、形态学梯度：

$$g = (f \oplus B) - (f \ominus B)$$

膨胀粗化一副图像中的区域，腐蚀则细化它们。膨胀和腐蚀的差强调了区域间的边界。同质区域不受影响，因此减操作趋于消除同质区域。最终结果是边缘

被增强而同质区域的贡献被抑制，从而产生类似于微分的效果。

## 四、 结果分析

### （一）二值图像

对于二值图像的处理，实验采用了全为 1 的 3\*3 的结构算子

#### 1、膨胀腐蚀

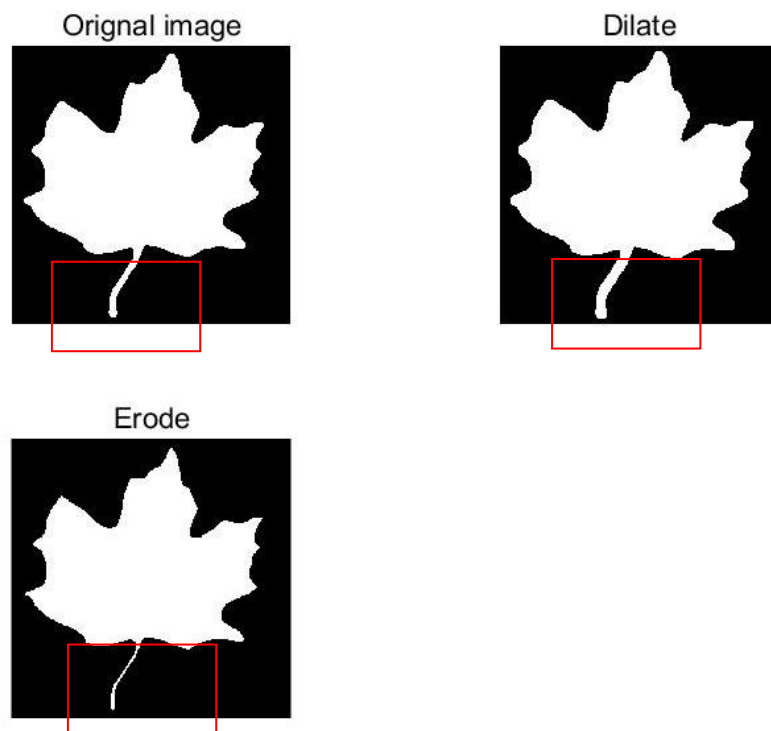


图 1 腐蚀和膨胀

采用的树叶的二值图，叶面经过处理后变化不明显，但是观察三幅的红色框中叶柄的变化，在叶柄的大概轮廓不变的前提下，叶柄经过膨胀处理后变粗了；腐蚀后的叶柄变细了。

#### 2、边缘提取

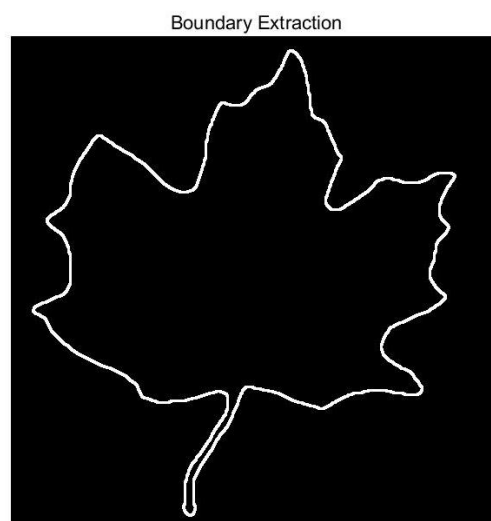


图 2 边缘提取

用原图减去腐蚀后的图像可得边缘提取结果，实验结果如图 2 所示。边缘即为树叶的样子。

### 3、交差补运算

另外选择了一个芯片图片（图 4），配合树叶图片进行交差补运算。

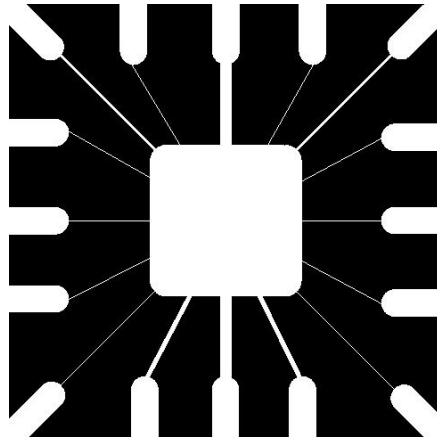


图 3 芯片原图

实验结果如图 4 所示，这是一些基本逻辑运算，常用于和腐蚀、膨胀配合在一起，实现开闭运算等等。

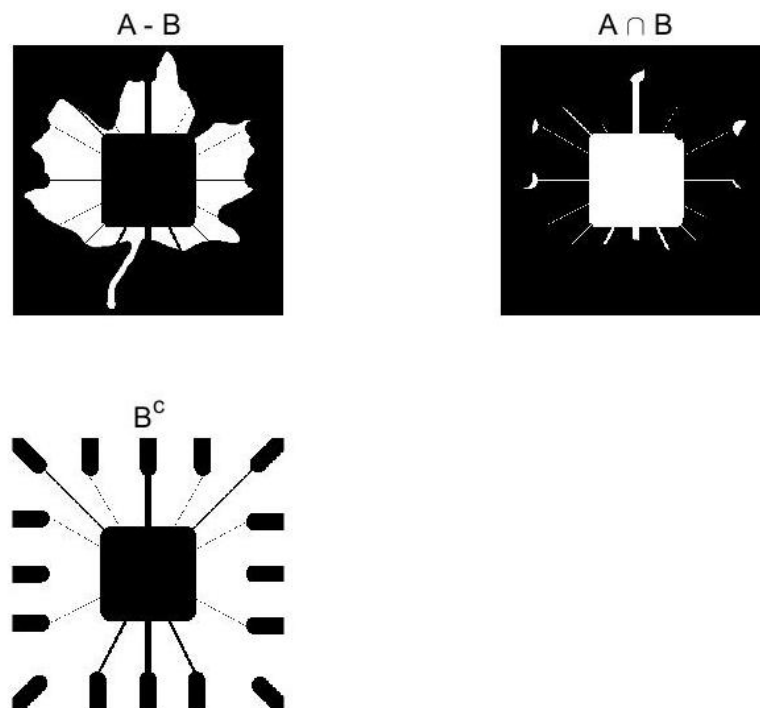


图 4 交差补

## （二）灰度图

处理灰度图像，任然选用和二值图像一样的结构元素。

### 1、 膨胀腐蚀

选用了一张五角星图片进行处理，五角星收到了不规则白色线条的污染。

膨胀处理后灰度图像的亮特征变浓了，白色线条变粗，人的视觉会感觉线条变得更亮。而暗的地方更暗，暗特征降低了。

与膨胀相反，腐蚀处理后灰度图像的暗特征变浓了，亮特征降低了，此幅图中，甚至将白色线条污染去除了。因此，腐蚀在一些场合中也可以用来去干扰。

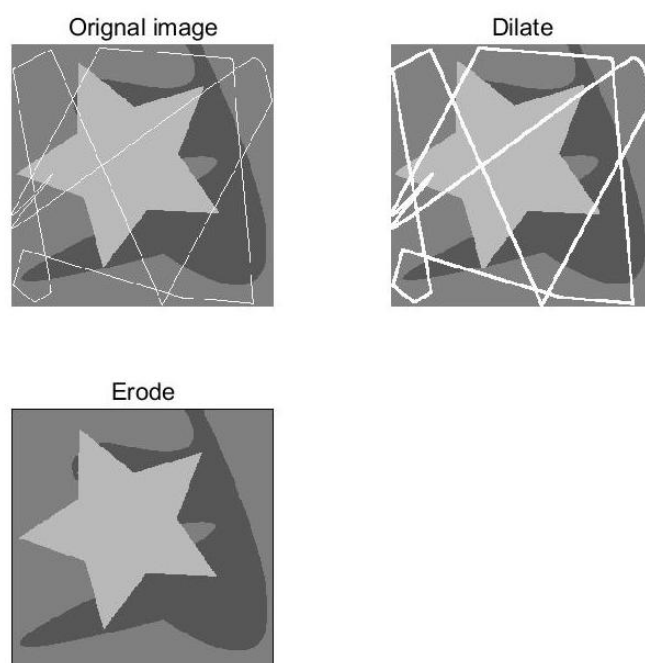


图 5 腐蚀膨胀

### 2、 交差补

原图选用视频监控中的两个截图，二者差别在于 A 图有一辆驶入画面中的轿车，B 图没有。



图 6 原图

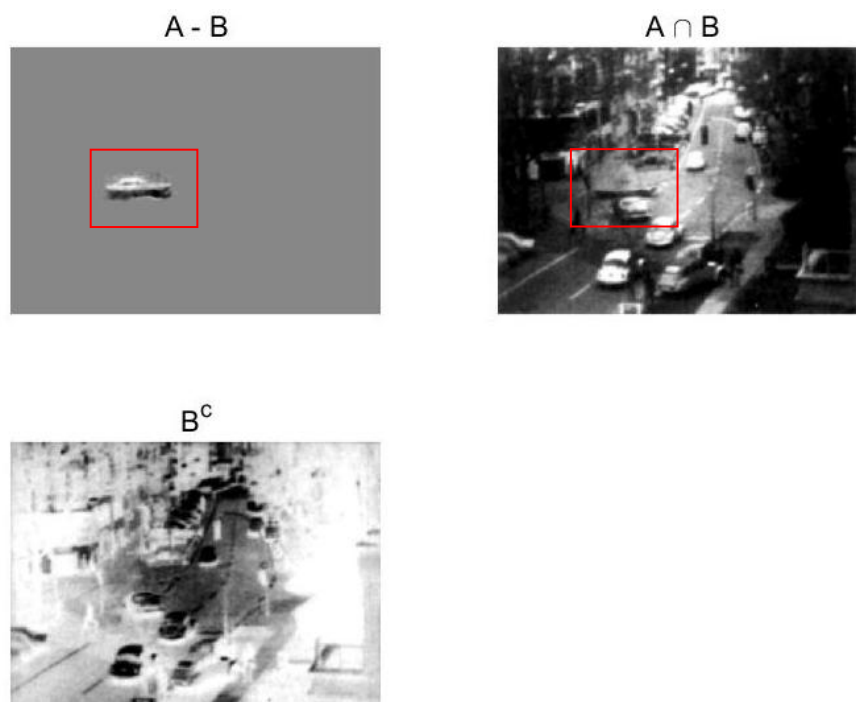
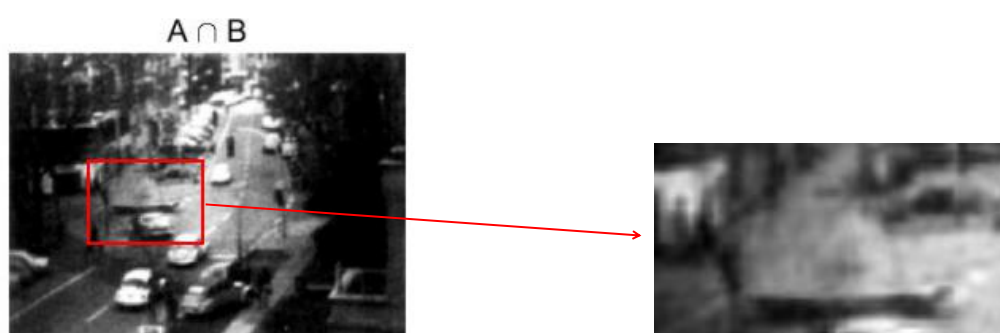


图 7 基本逻辑运算

经过差运算，可以检测出这辆与众不同的轿车，差运算是图像处理中帧差法的基本操作，在有视频信息的情况下，可以选取某些帧进行背景建模，然后一旦监控画面有改变，出现了某个人、某辆车，画面改变，可以用当前帧减去背景帧，实现人像的检测等等。

交运算是每个像素点选取两幅灰度图像中该点值最小的一个。车的上半部分比较白，灰度值高，未被保留；下半部分，灰度值低，几乎是黑色，被保留在二者相交的结果中：



### 3、形态学梯度 VS 简单边界提取

经过形态学梯度公式处理后的图像边缘被增强而同质区域的贡献被抑制，见图 7（a），产生了类似于微分的效果。简单的用“原图-腐蚀图像”进行边界提取如图 7（b）所示，二者对比会发现利用形态学梯度进行边界提取保留了更多细节，而且边界提取的结果更精准。

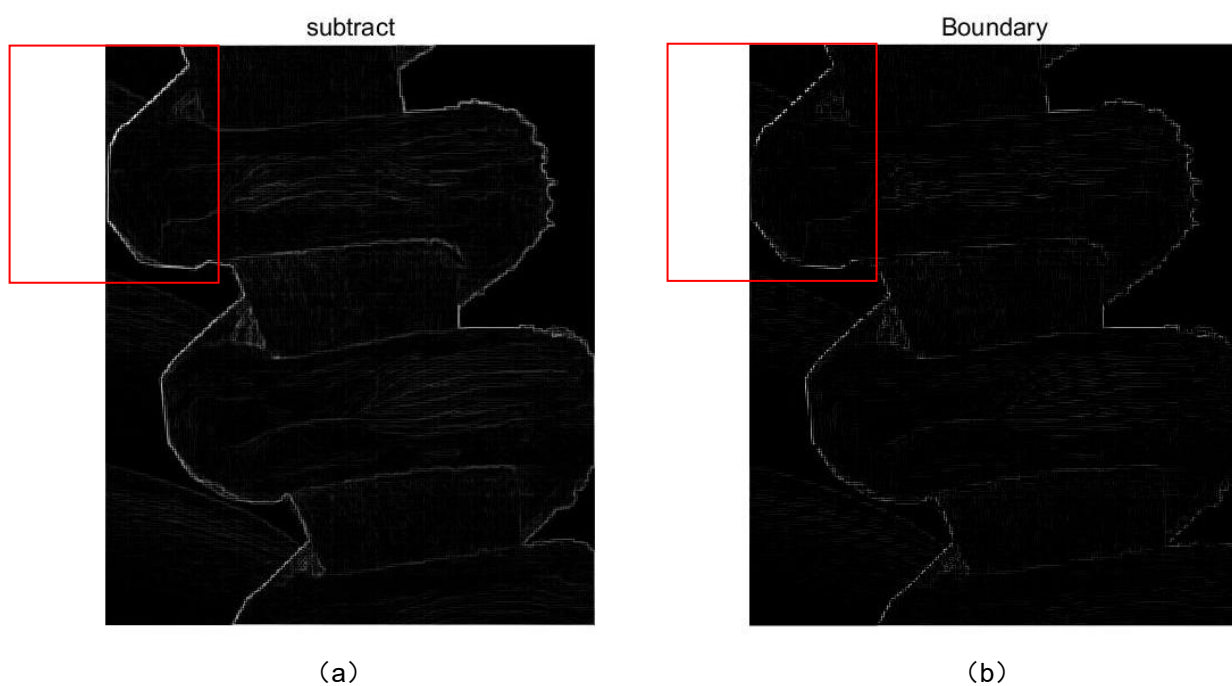


图 8 边界提取对比

实验中也对改图进行了开运算，但是由于图片不典型，因此开运算后的结果和原图差别不大。

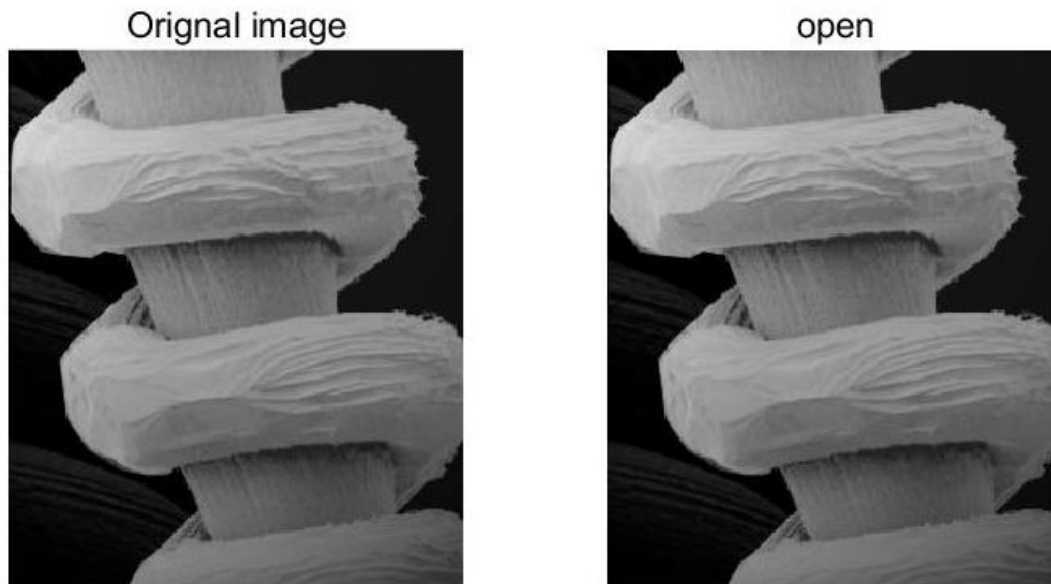


图 9 开运算