

浙江大学实验报告

专业：信息工程

姓名：徐晓刚

学号：3140102480

日期：2017 /5 /9

地点：

课程名称：数字图像处理实验 指导老师：项志宇 成绩：

实验名称：灰度级形态学处理 实验类型：探究型 同组学生姓名：无

一、实验目的和要求（必填）

二、实验内容和原理（必填）

三、主要仪器设备（必填）

四、操作方法和实验步骤

五、实验数据记录和处理

六、实验结果与分析（必填）

七、讨论、心得

一、实验目的和要求

1. 掌握灰度级形态学处理方法，如腐蚀，膨胀等基本操作；
2. 使用灰度级形态学的方法计算一副图像的形态学梯度。

二、实验内容和原理

1) 实验原理

1. 灰度级形态学腐蚀

我们在这里定义高度为 1，对称且平坦的结构元 b ，其原点位于中心处。即其满足的是

$$\hat{b}(x, y) = b(-x, -y)$$

用一个平坦的结构元 b 在 (x, y) 处对图像 f 的腐蚀定义为图像 f 中与 b 重合区域的最小值。其公式表示结构元素对一幅图像 f 在位置 (x, y) 处的腐蚀由下式给出：

$$[f \ominus b](x, y) = \min_{(s, t) \in b} \{f(x + s, y + t)\}$$

也就是说为寻找 b 对 f 的腐蚀，我们可以把结构元的原点放在图像每一个像素的位置，在任何位置的腐蚀由从包含在与 b 重合区域中的 f 的所有值中选取的最小值决定。在这里实现的时候，我们假定 b 定义在 3×3 区域，所以每一个点的腐蚀要求寻找在这个区域中的 f 的 9 个值中其原点所在的那个点的最小值。

2. 灰度级形态学膨胀

用一个平坦的结构元 b 在 (x, y) 处对图像 f 的膨胀定义为图像 f 中与 b 重合区域的最大值。其公式表示结构元素对一幅图像 f 在位置 (x, y) 处的膨胀由下式给出：

$$[f \oplus b](x, y) = \max_{(s, t) \in b} \{f(x - s, y - t)\}$$

在这里由于 b 是平坦的，也就是其所有元素都是关于其原点对称的。那么我们为寻找 b 对 f 的膨胀，我们可以把结构元的原点放在图像每一个像素的位置，在任何位置的膨胀由从包含在与 b 重合区域中的 f 的所有值中选取的最大值决定。在这里实现的时候，我们假定也 b 定义在 3×3 区域，所以每一个点的膨胀要求寻找在这个区域中的 f 的 9 个值中其原点所在的那个点的最大值。

3. 形态学梯度

膨胀和腐蚀可与图像相减结合起来得到一幅图像的形态学梯度，在这里，我们由 g 定义：

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

膨胀粗化一幅图像中的区域，而腐蚀操作则会将它们进行细化。膨胀和腐蚀的差强调了区域

之间的边界。在这里只要我们设定的结构元的大小不是很大，同质区域不受到影响。因此相减的操作可以产生消除同质区域的效果。最终的结果是边缘被增强而同质区域的贡献被抑制。从而我们可以看到，这一操作就产生了类似于微分的操作，也就是得到了图像的梯度。在这里这一操作就通常被称为是形态学梯度。

2) 实验内容

掌握灰度级形态学处理方法，如腐蚀，膨胀等基本操作，并且使用 **Matlab** 编程实现其函数。根据以上所述的原理，使用灰度级形态学的方法计算一副图像的形态学梯度。

三、主要仪器设备

计算机，**Matlab**

四、操作方法和实验步骤

具体的代码实现采用 **Matlab** 编程环境

五、实验数据记录和处理

在这里我们使用的原始灰度图像如下所示：

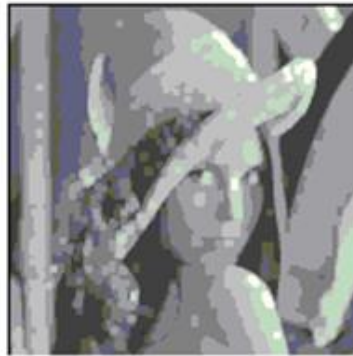


1. 首先我们需要使用编程实现对图像进行膨胀操作的函数。

在这里我们使用的结构元为高度为 1，对称且平坦的结构元 **b**，其定义的区域大小为 3×3 。所以每一个点的膨胀要求寻找在这个区域中的 **f** 的 9 个值中其原点所在的那个点的最大值。我们使用 **matlab** 代码的编写如下：

```
function dstimg = dilate(srcimg)
if length(srcimg) == 3
    srcimg = rgb2gray(srcimg);
end
[m, n] = size(srcimg);
for i = 2:m-1
    for j = 2:n-1
        neighbor = [srcimg(i,j), srcimg(i,j-1), srcimg(i,j+1)...
                    srcimg(i+1,j), srcimg(i+1,j-1), srcimg(i+1,j+1)...
                    srcimg(i-1,j), srcimg(i-1,j-1), srcimg(i-1,j+1)];
        dstimg(i,j) = max(neighbor);
    end
end
end
```

其对图像产生的效果如下：



可以看到膨胀的效果使得亮特征变浓了，而暗特征降低了。这符合我们平常的膨胀操作效果，故而可以认为我们的膨胀函数编写是正确的。

2. 我们需要使用编程实现对图像进行腐蚀操作的函数。

在这里我们使用的结构元为高度为 1，对称且平坦的结构元 b ，其定义的区域大小为 3×3 。所以每一个点的腐蚀要求寻找在这个区域中的 f 的 9 个值中其原点所在的那个点的最小值。我们使用 matlab 代码的编写如下：

```
function dstimg = erode(srcimg)
if length(srcimg) == 3
    srcimg = rgb2gray(srcimg);
end
[m, n] = size(srcimg);
for i = 2:m-1
    for j = 2:n-1
        neighbor = [srcimg(i,j), srcimg(i,j-1), srcimg(i,j+1)...
                    srcimg(i+1,j), srcimg(i+1,j-1), srcimg(i+1,j+1)...
                    srcimg(i-1,j), srcimg(i-1,j-1), srcimg(i-1,j+1)];
        dstimg(i,j) = min(neighbor);
    end
end
end
```

对图像产生的效果如下所示：

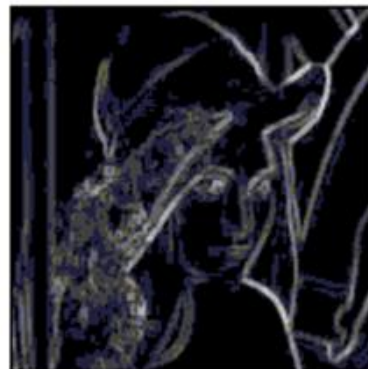


可以看到腐蚀的效果使得暗特征变浓了，而亮特征降低了。这符合我们平常的腐蚀操作效果，故而可以认为我们的腐蚀函数编写是正确的。

3. 最后我们可以用以上编写的膨胀函数和腐蚀函数来实现形态学梯度的计算，计算的公式为：

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

最后的效果如下所示：



可以看到区域之间的边界被清楚地描绘出来了，这与二维微分图像的预期是相同的。原本的同质区域都已经成为黑色。

六、实验结果与分析

实验中通过使用形态学的结构元方法，定义和编写了对图像进行膨胀和腐蚀操作的函数。从其中可以看出膨胀的效果是在图像中增加亮特征的浓度，并且降低暗特征；腐蚀的效果是在图像中增加暗特征的浓度并且降低亮特征。在此基础上，我们可以使用膨胀和腐蚀的差强调了区域之间的边界的原理，使用两者相减来获得在一幅图像中的梯度。在这里会产生类似于图像微分产生的梯度效果。

最后的实验结果表面我们的编程实现是正确的。

七、讨论、心得

本次实验的原理性是比较陌生的。主要需要我们先去掌握形态学处理的知识。在形态学的处理中，通过设定结构元的特征，并且设定其对图像的操作，可以实现在之前使用过的多种滤波器的效果。但是这种多是在空间域上的处理。在这次的实验中，以梯度和边界的提取为例，首先在灰度级实现了对图像进行膨胀的操作，以及对图像进行腐蚀的操作。由此我们可以实现形态学梯度的提取。效果表明其与我们直接使用边缘检测算子，比如拉普拉斯算子，效果是差不多的，我们最后可以得到比较好的梯度边缘效果。