

机器视觉技术第四次作业

1913416 阎丝雨

一、 算法简介与代码实现

(一) 实现图像的灰度级膨胀与灰度级腐蚀

实现原理: 腐蚀运算是由结构元素确定的邻域块中选取图像值与结构元素值的差的最小值。膨胀运算是由结构元素确定的邻域块中选取图像值与结构元素值的和的最大值

➤ 输入图像并调用自定义膨胀与腐蚀函数

```
I=im2double(imread('small-blobs.tif'));
figure(1), subplot(1,3,1), imshow(I), title('原图');
B=[1 1 1;1 1 1;1 1 1];
J_Erosion=corrode(I,B);
J_Dilation=expand(I,B);
subplot(1,3,2), imshow(J_Erosion), title('腐蚀');
subplot(1,3,3), imshow(J_Dilation), title('膨胀');
```

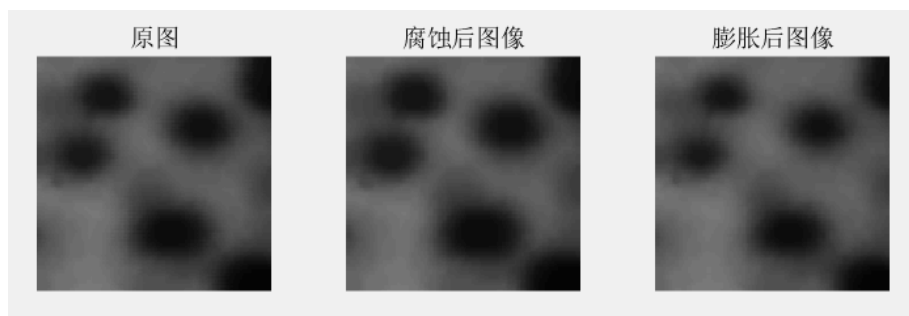
➤ 腐蚀函数

```
function [J_Erosion] = corrode(I,B)
n=size(B,1);
ind=B==0;
n_1=floor(n/2);
%对边界图进行扩充，目的是为了处理边界点
I_pad=padarray(I,[n_1,n_1],'replicate');
[M,N]=size(I);
J_Erosion=zeros(M,N);
for i=1:M
    for j=1:N
        %获得图像子块区域
        Block=I_pad(i:i+2*n_1,j:j+2*n_1);
        C=Block.*B;
        %删除0值，保留4连通数值
        C=C(:);
        C(ind)=[];
        %腐蚀操作
        J_Erosion(i,j)=min(C);
    end
end
end
```

➤ 膨胀函数

```
function [J_Dilation] = expand(I,B)
n=size(B,1);
ind=B==0;
n_1=floor(n/2);
%对边界图进行扩充，目的是为了处理边界点
I_pad=padarray(I,[n_1,n_1],'replicate');
[M,N]=size(I);
J_Dilation=zeros(M,N);
for i=1:M
    for j=1:N
        %获得图像子块区域
        Block=I_pad(i:i+2*n_1,j:j+2*n_1);
        C=Block.*B;
        %删除0值，保留4连通数值
        C=C(:);
        C(ind)=[];
        %膨胀操作
        J_Dilation(i,j)=max(C);
    end
end
end
```

➤ 运行结果



(二) 对梯度幅值图像进行形态学平滑

由于直接给出梯度幅值图像，故无需对小气泡图进行处理，仅在此对得到梯度幅值图像处理过程做简单介绍：因为图像是离散函数，因此在求梯度时无法对其进行求导，故选取中心差商法获取某点的导数值，可以选用 Sobel 算子对图像进行卷积得到 x 两个方向的梯度，再进行勾股定理得出梯度幅值图。

形态学平滑处理表现为对图像先进性开运算后进行闭运算，目的是消除分水岭分割过分割问题。

开运算：先进行腐蚀操作，再进行膨胀操作；

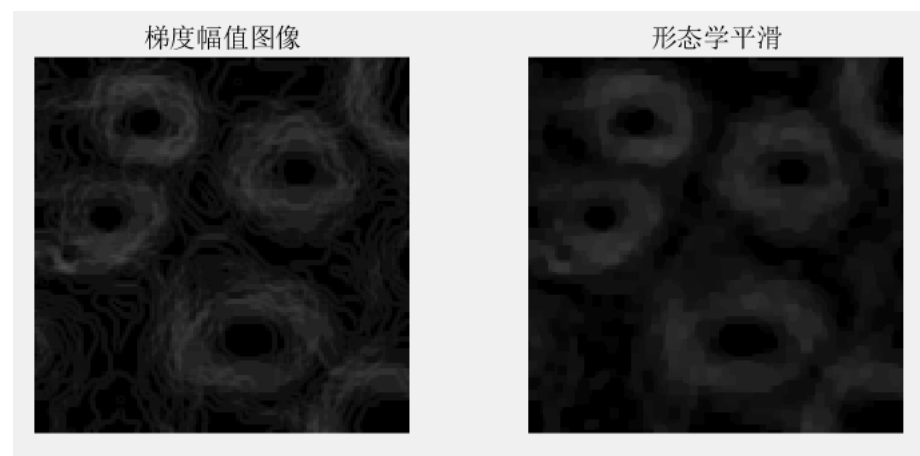
闭运算：先进行膨胀操作，再进行腐蚀操作。

本段代码直接调用接口函数实现开、闭运算，若不调用 Matlab 内置函数也可按照腐蚀-膨胀-膨胀-腐蚀的顺序来调用本程序中的自定义函数。

➤ 实现代码如下

```
I_2=imread('small-blobs-gradient.tif');  
figure(2), subplot(1,2,1), imshow(I_2), title('梯度幅值图像');  
%开运算  
I_3=imopen(I_2,B);  
%闭运算  
I_3=imclose(I_3,B);  
subplot(1,2,2), imshow(I_3), title('形态学平滑')
```

➤ 运行结果



(三) 分水岭分割

分水岭算法是一种图像区域分割法，在分割的过程中，它会把跟临近像素间的相似性作为重要的参考依据，从而将在空间位置上相近并且灰度值相近的像素点互相连接起来构成一个封闭的轮廓，封闭性是分水岭算法的一个重要特征。

本段代码首先调用 `watershed()` 函数实现分水岭算法，之后调用 `label2rgb()` 函数将其变为 RGB 彩色图像，再调用 `rgb2gray()` 将图片转为灰度图像。对原始梯度幅值图像和平滑后的梯度幅值图像都进行这样的处理。

➤ 实现代码如下

```
%对原始梯度幅值图像进行分水岭分割
I_4=watershed(I_2);
I_4=label2rgb(I_4);%并将生成的标签矩阵显示为RGB图像
I_4=rgb2gray(I_4);
figure(3);
subplot(1,2,1),imshow(I_4),title('原始梯度幅值图像的分水岭分割结果')

%对平滑后的梯度幅值图进行分水岭分割
I_5=watershed(I_3);
I_5=label2rgb(I_5);%并将生成的标签矩阵显示为RGB图像
I_5=rgb2gray(I_5);
subplot(1,2,2),imshow(I_5),title('平滑后的梯度幅值图的分水岭分割结果')
```

➤ 运行结果

