1 边缘检测

在图像中，边缘是图像局部强度变化最明显的地方，它主要存在于目标与目标、目标与背景、区域与区域( 包括不同色彩) 之间。边缘表明一个特征区域的终结和另一特征区域的开始。边缘所分开区域的内部特征或属性是一致的，而不同的区域内部特征或属性是不同的。边缘检测正是利用物体和背景在某种图像特征上的差异来实现检测，这些差异包括灰度、颜色或纹理特征，边缘检测实际上就是检测图像特征发生变化的位置。边缘的类型很多，常见的有以下三种: 第一种是阶梯形边缘，其灰度从低跳跃到高; 第二种是屋顶形边缘，其灰度从低逐渐到高然后慢慢减小; 第三种是线性边缘，其灰度呈脉冲跳跃变化。

2 微分边缘检测算子

2.1一阶微分算子

2.1.1 Roberts 算子

Roberts 算子是一种利用局部差分算子寻找边缘的算子, 它由下式给出：

 (1)

其中，、、和分别为4领域的坐标，且是具有整数像素坐标的输入图像。

Robert算子是算子模板。图2所示的2个卷积核形成了Roberts算子。图像中的每一个点都用这2个核做卷积。

|  |  |
| --- | --- |
| 1 | 0 |
| 0 | -1 |

|  |  |
| --- | --- |
| 0 | 1 |
| -1 | 0 |

图2 Roberts算子

Roberts 算子边缘定位精度较高，但容易丢失一部分边缘，同时由于没有经过图像平滑计算，因此不能抑制噪声，该算子对具有陡峭的低噪声图像响应较好。

2.1.2 Sobel算子

Sobel算子是一种一阶微分算子，它利用像素临近区域的梯度值来计算1个像素的梯度，然后根据一定的阈值来取舍。它由下式给出：

 (2)

Sobel算子是算子模板。图3所示的2个卷积核、形成Sobel算子。一个核通常的垂直边缘响应最大，而另一个核对水平边缘响应最大。2个卷积的最大值作为该点的输出值。运算结果是一幅边缘幅度图像。

|  |  |  |
| --- | --- | --- |
| -1 | 0 | 1 |
| -2 | 0 | 2 |
| -1 | 0 | 1 |

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 1 |
| 0 | 0 | 0 |
| -1 | -2 | -1 |

图3 Sobel算子

2.1.3 Prewitt 算子

Prewitt 算子由下式给出：

 (3)

Prewitt算子是算子模板。图4所示的2个卷积核和形成了Prewitt算子。与Sobel算子的方法一样，图像中的每个点都用这2个核进行卷积。取最大值作为输出值。Prewitt算子也产生一副边缘幅度图像。

|  |  |  |
| --- | --- | --- |
| -1 | 0 | 1 |
| -1 | 0 | 1 |
| -1 | 0 | 1 |

|  |  |  |
| --- | --- | --- |
| 1 | 1 | 1 |
| 0 | 0 | 0 |
| -1 | -1 | -1 |

图4 Prewitt算子

2.1.4 Canny 算子

传统的Canny算法是通过在邻域内求有限差分来计算梯度幅值。Canny算子法实现的方式为：图像先用2D高斯滤波模板进行卷积以消除噪声，再对滤波后图像中的每个像素计算其梯度的大小和方向。计算可采用以下大小的模板作为对方向和方向偏微分的一阶近似：

由此得到梯度的大小M和方向：





通过梯度的方向，可以找到这个像素梯度方向的邻接像素：

|  |  |  |
| --- | --- | --- |
| 3 | 2 | 1 |
| 0 | x | 0 |
| 1 | 2 | 3 |

最后通过非最大值抑制以及阈值化和边缘连接。Canny算子有信噪比准则、定位精度准则和单边缘响应准则。Canny算法的实质是用一个准高斯函数做平滑运算，然后以带方向的一阶微分算子定位导数最大值，它可用高斯函数的梯度来近似，在理论上很接近k 个指数函数的线性组合形成的最佳边缘算子。它是一阶传统微分中检测阶跃性边缘效果最好的算子之一，它比Prewitt 算子、Sobel 算子的去噪能力都要强，但它也容易平滑掉一些边缘信息，其检查方法较为复杂。

2.2 二阶微分算子

2.2.1 Laplacian 算子

拉普拉斯二阶零交叉(zerocross)算子是利用边缘点处二阶导函数出现零交叉原理来检测边缘。函数的拉普拉斯算子公式为：



使用差分方程对x和y方向上的二阶偏导数近似如下：







这一近似式是以点[i,j+1]为中心的。用j-1替换j,得到以点[i,j]为中心的二阶偏导数的理想近似式：

 (1)

类似地，可得：

 (2)

把(1)、(2)式合并可得能用来近似表达拉普拉斯算子的模板：



当拉普拉斯算子输出出现过零点时就表明有边缘存在，其中忽略无意义的过零点(均匀零区) 。原则上，过零点的位置精度可以通过线性内插方法精确到子像素的分辨率，不过由于噪声，结果可能不会很精确。拉普拉斯算子不具方向性，对灰度突变敏感，定位精度较高，同时，对噪声也敏感。Laplacian算子一般不以其原始形式用于边缘检测。

2.2.2 LOG 算子

正如上面所提到的，利用图像强度二阶导数的零交叉点来求边缘点的算法对噪声十分敏感，所以，希望在边缘增强前滤除噪声。将高斯滤波器和拉普拉斯零交叉算子结合在一起就形成了Log 算子。Log 算子实现的方式有两种: 一种是图像先与高斯滤波器进行卷积，再求卷积的拉普拉斯变换; 另一种是先求高斯滤波器的拉普拉斯变换，再求与图像的卷积。Log 边缘检测器的基本特征是: ①平滑滤波器是高斯滤波器; ②增强步骤采用二阶导数(二维拉普拉斯函数) ; ③边缘检测判据是二阶导数零交叉点并对应一阶导数的较大峰值; ④使用线性内插方法在子像素分辨率水平上估计边缘的位置。Log算子的输出可通过卷积运算得到：



根据卷积求导法有：



其中，

Log 算子法既平滑了图像又降低了噪声，由于平滑会导致边缘的延展，因此边缘检测器只考虑那些具有局部梯度最大值的点为边缘点，这一点可以用二阶导数的零交叉点来实现。为了避免检测出非显著边缘，选择一阶导数大于某一阈值的零交叉点作为边缘点。

3 MATLAB设计

3.1 程序说明

3.1.1 滤波

进行3x3的均值滤波操作

n = 3;

template = ones(n);

[height, width] = size(I0);

x1 = double(I0);

x2 = x1;

for i = 1:height-n+1

for j = 1:width-n+1

c = x1(i:i+n-1,j:j+n-1).\*template;

s = sum(sum(c));

x2(i+(n-1)/2,j+(n-1)/2) = s/(n\*n);

end

end

3.1.2 边缘检测

用matlab自带的边缘检测函数，分别进行Roberts、Sobel、Prewitt、LOG、Canny检测

通用方法为：Edge(原始图片，边缘检测方法，显示阈值)

BW1=edge(I0,'Roberts',0.016);

subplot(232);

%figure

imshow(BW1);

title('Robert算子边缘检测')

BW2=edge(I0,'Sobel',0.016);

subplot(233);

%figure

imshow(BW2);

title('Sobel算子边缘检测')

BW3=edge(I0,'Prewitt',0.016);

subplot(234);

%figure

imshow(BW3);

title('Prewitt算子边缘检测');

BW4=edge(I0,'LOG',0.001);

subplot(235);

%figure

imshow(BW4);

title('LOG算子边缘检测')

BW5=edge(I0,'Canny',0.02);

subplot(236);

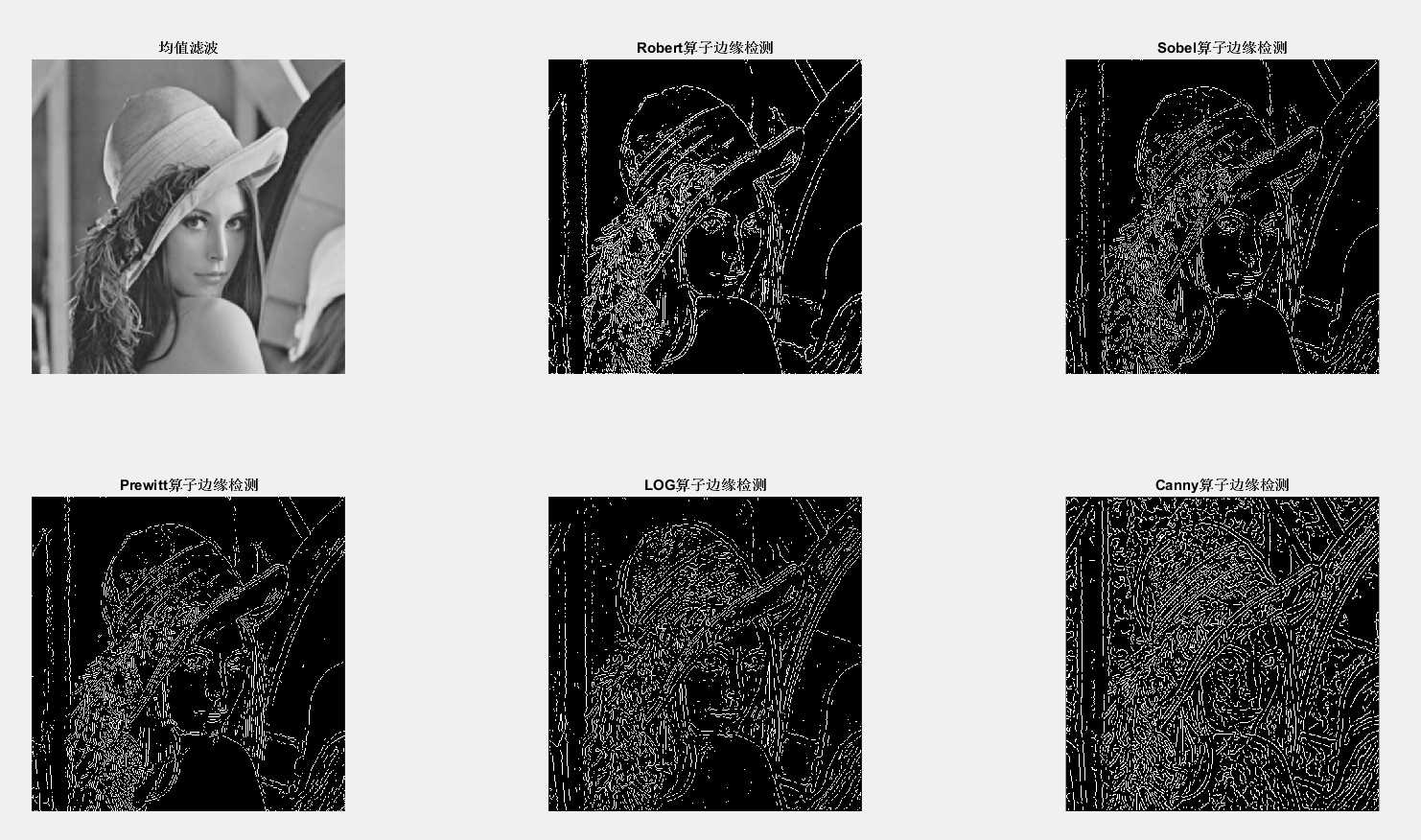
%figure

imshow(BW5);

title('Canny算子边缘检测')

3.1 结果展示

MATLAB图像处理工具中有多种边缘检测算子函数，利用MATLAB edge 函数，得以下实验结果：



各算子的MATLAB 检测函数均未带滤波器，阈值为观察到检测结果为最优的阈值。从以上可以看出

* Robert：边缘定位精度较高，对于陡峭边缘且噪声低的图像效果较好，但没有进行平滑处理，没有抑制噪声的能力。
* sobel和prewitt：进行了平滑处理，对噪声具有一定抑制能力，但容易出现多像素宽度。
* Laplacian：对噪声较为敏感，使噪声能力成分得到加强，容易丢失部分边缘方向信息，造成一些不连续的检测边缘，同时抗噪声能力较差。
* log：抗噪声能力较强，但会造成一些尖锐的边缘无法检测到。
* canny：最优化思想的边缘检测算子，同时采用高斯函数对图像进行平滑处理，但会造成将高频边缘平滑掉，造成边缘丢失，采用双阈值算法检测和连接边缘。

。

4 结束语

根据以上实验和算法分析可得，Roberts，Sobel和Prewitt 算子的算法较为简单，容易实现，运算速度较快，对噪声敏感，可用于车牌号码识别、流水线上产品检测、电视节目字幕检测等对识别速度要求较高而对精度要求不高的地方。Log算子的算法稍微复杂一些，其检测效果好于Roberts,Sobel 和Prewitt算子，可用于答卷识别、邮政分捡等对识别速度和精度都有一定要求的地方。改进的Sobel梯度边缘检测算法，虽然运行时间较原算法多一些，但是它克服了sobel算子进行边缘检测存在边缘粗糙、对噪声敏感的缺点，具有提取的边缘精细、抗噪能力强等优点，是一种简单有效的边缘检测算法。Canny算子算法最为复杂，但其检测效果为最好，可用于医学识别、遥测等对速度要求不高而对精度要求较高的地方。在应用中，应根据实际情况选择不同的微分算子。

附：

1. matlab边缘检测程序

%读入图片

I=imread('2.png');

I0=rgb2gray(I);

%均值滤波

n = 3;

template = ones(n);

[height, width] = size(I0);

x1 = double(I0);

x2 = x1;

for i = 1:height-n+1

for j = 1:width-n+1

c = x1(i:i+n-1,j:j+n-1).\*template;

s = sum(sum(c));

x2(i+(n-1)/2,j+(n-1)/2) = s/(n\*n);

end

end

g = uint8(x2);

subplot(231);

%figure

imshow(g)

title('均值滤波')

I0=g

BW1=edge(I0,'Roberts',0.016);

subplot(232);

%figure

imshow(BW1);

title('Robert算子边缘检测')

BW2=edge(I0,'Sobel',0.016);

subplot(233);

%figure

imshow(BW2);

title('Sobel算子边缘检测')

BW3=edge(I0,'Prewitt',0.016);

subplot(234);

%figure

imshow(BW3);

title('Prewitt算子边缘检测');

BW4=edge(I0,'LOG',0.001);

subplot(235);

%figure

imshow(BW4);

title('LOG算子边缘检测')

BW5=edge(I0,'Canny',0.02);

subplot(236);

%figure

imshow(BW5);

title('Canny算子边缘检测')