

数字图像处理

直线检测

姓名：余静

班级：自动化 62

学号：2160504052

提交日期：2019 年 5 月 18 日

摘要

一条线可视为一条边缘线段，该线两侧的背景灰度要么远亮于该线的像素的灰度，要么远暗于。孤立点可视为一条线，只是其长度和宽度都为一个像素。本次作业主要基于孤立点，线以及边缘检测 在MATLAB 上实现了边缘检测以及直线检测。

1. 首先对测试图像（文件名为：test1~test6）进行边缘检测，可采用书上介绍的Sobel等模板或者cann算子方法；

1.1

- 1) Sobel 模板：

$$g_x = \frac{\partial f}{\partial x} = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$

<http://blog.csdn.net/TonyShengTan>

$$g_y = \frac{\partial f}{\partial y} = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$$

其下降速率（梯度的长度）计算公式：

$$M(x, y) \approx |(z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)| + |(z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)|$$

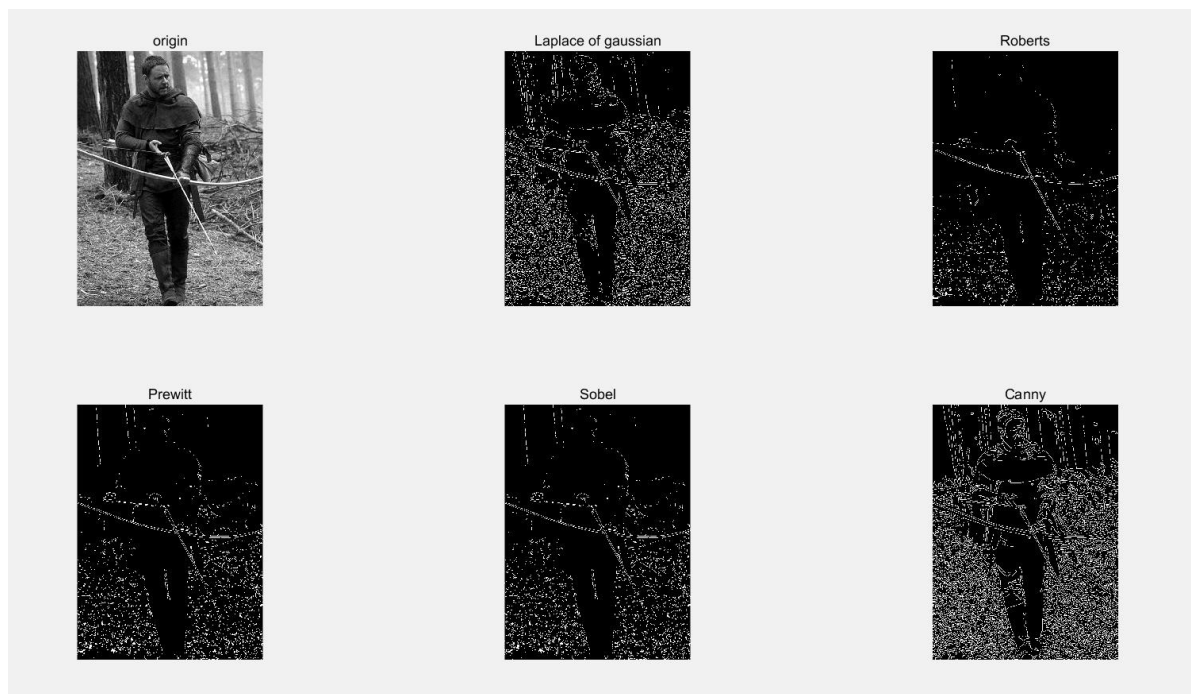
- 2) canny 算子：

用高斯滤波器平滑图像
用一阶偏导有限差分计算梯度幅值和方向

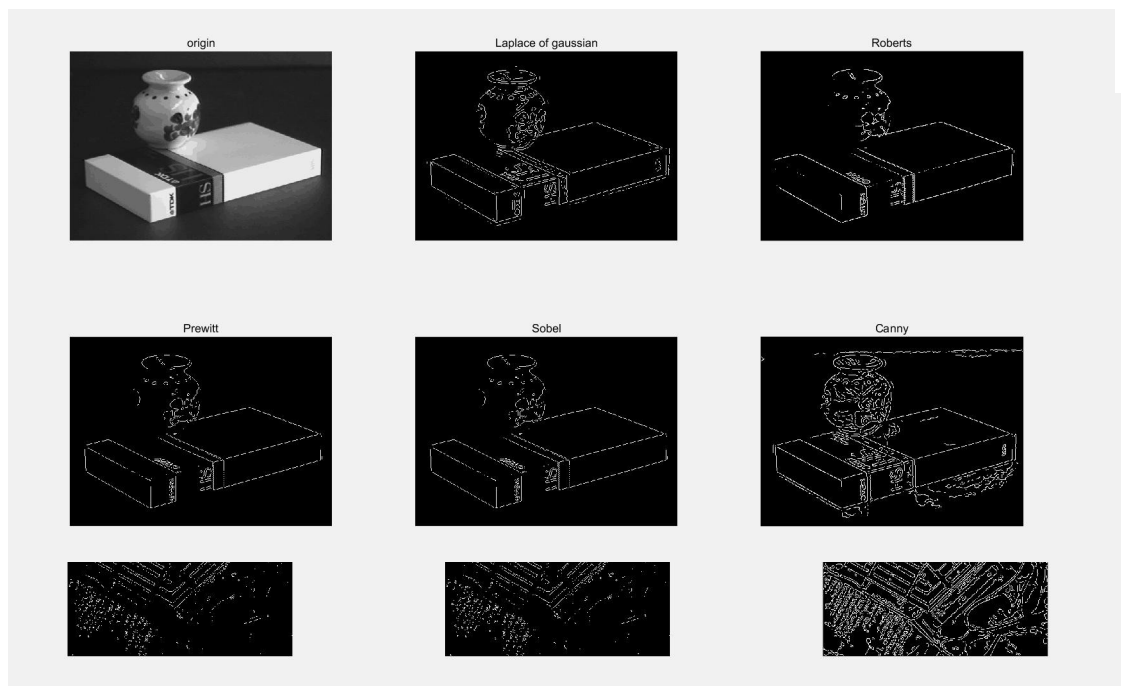
对梯度幅值进行非极大值抑制

用双阈值算法检测和连接边缘

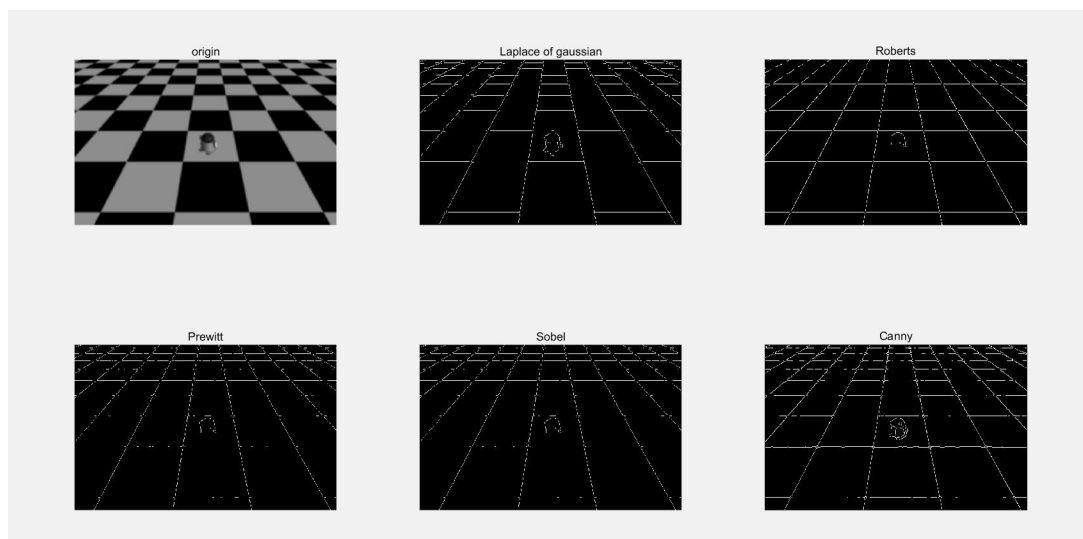
1.2 实验结果



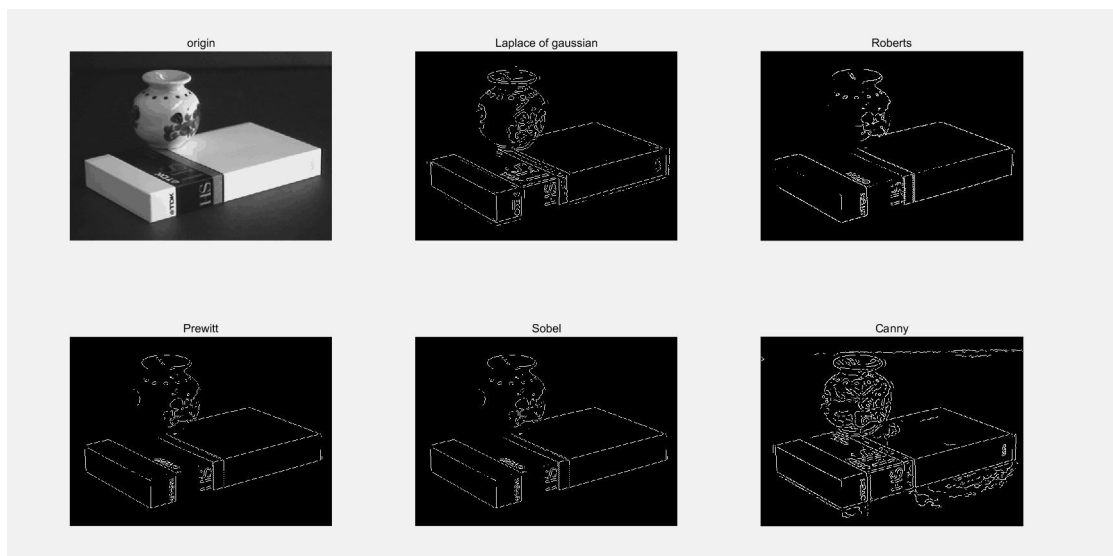
Test1



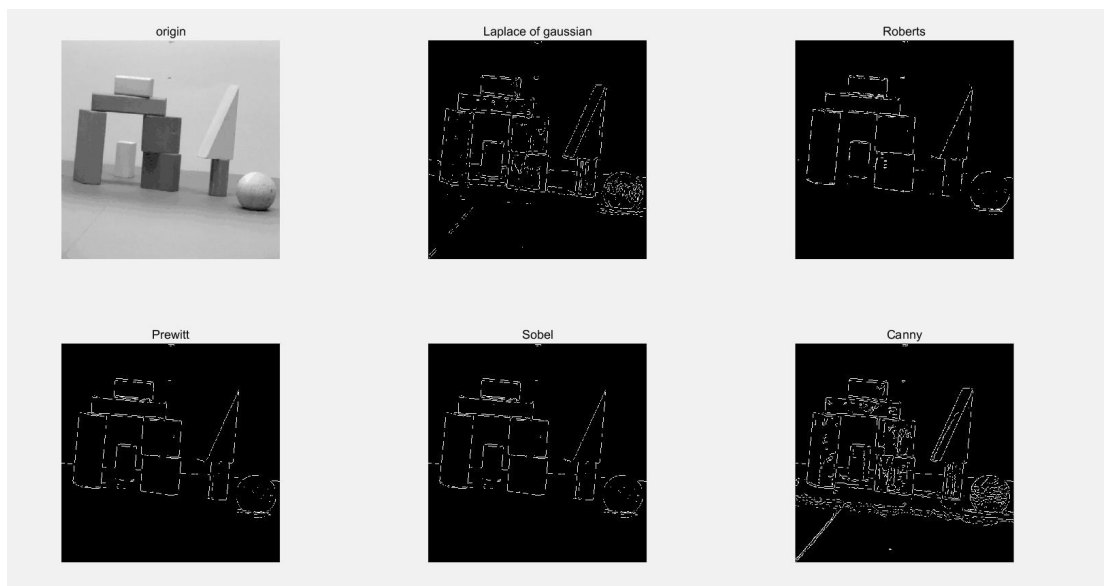
Test2



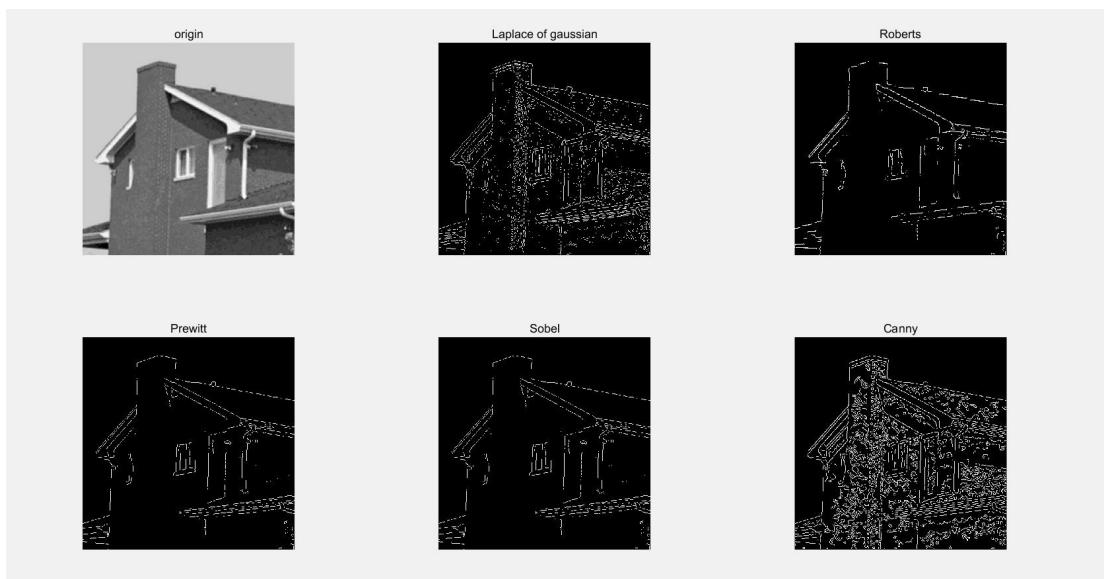
Test3



Test4



Test5



Test6

2. 在边缘检测的基础上，用hough变换检测图中直线；

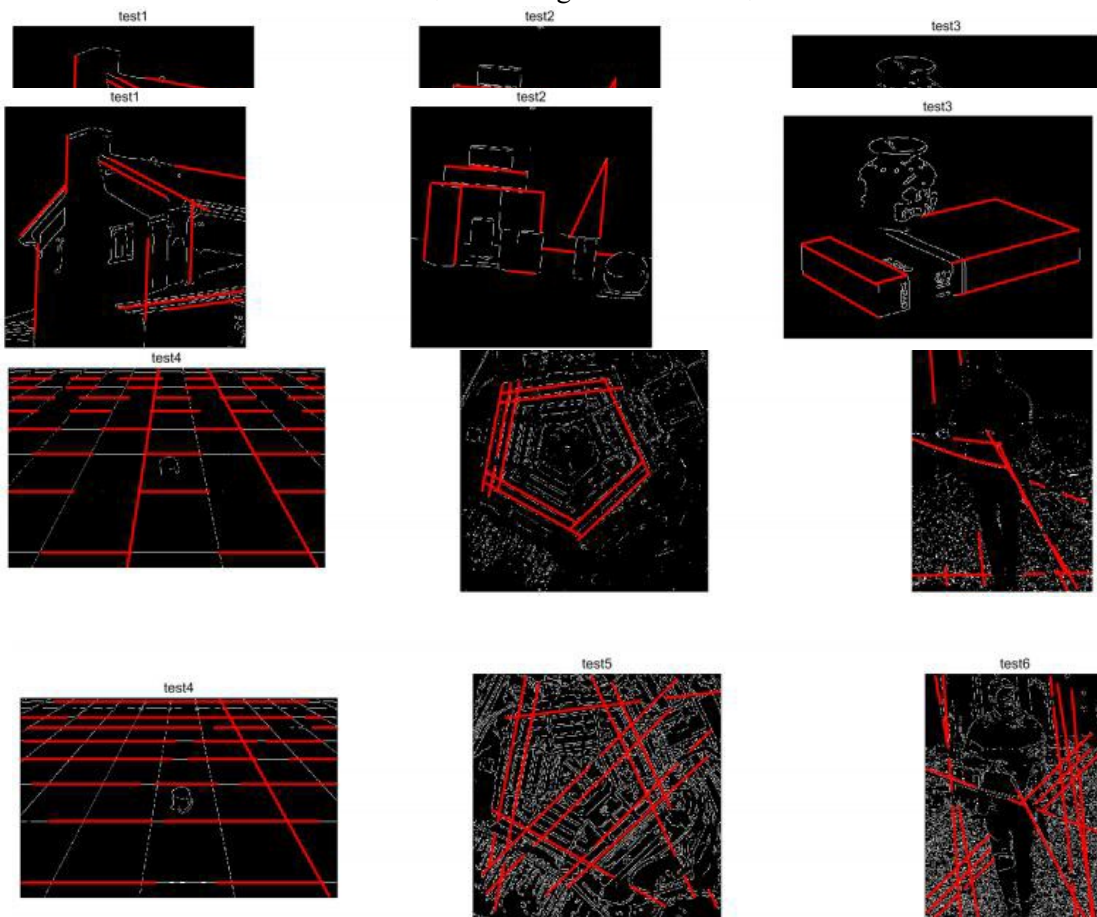
2.1 hough 变换原理：

$x-y$ 坐标系中直线方程为 $y=ax+b$ $x-y$ 坐标系中的每一个点在, 在 $a-b$ 坐标系中对应一条直线 $x-y$ 坐标系中共线的点集, 在 $a-b$ 坐标系中对应一个线束（射影几何中的概念），线束的中心 (a',b') , 在 $x-y$ 平面中可以确定一条直线. 但是直线方程 $y=ax+b$ 不能表示斜率无限大的直线，所用改用直线方程： $x\cos\theta+y\sin\theta=\rho$ $x-y$ 坐标系上的每一点在参数空间 $\theta-\rho$ 对应一条正弦线 $x-y$ 坐标系上共线的点集, 在 $\theta-\rho$ 空间中有一组正弦线，并且交于一点 (θ',ρ') , 在 $x-y$ 平面中可确定一条直线将图像空间（ $x-y$ 平面）中所有点映射到 $\theta-\rho$ 空间（实际上是二维矩阵），这个二维矩阵叫做“累计数组”或“累加器”。累计数组中值较大的元素，往往对应着图像中的直线。

2.2实验结果

基于 Sobel 和 Canny 算子的 Hough 变换

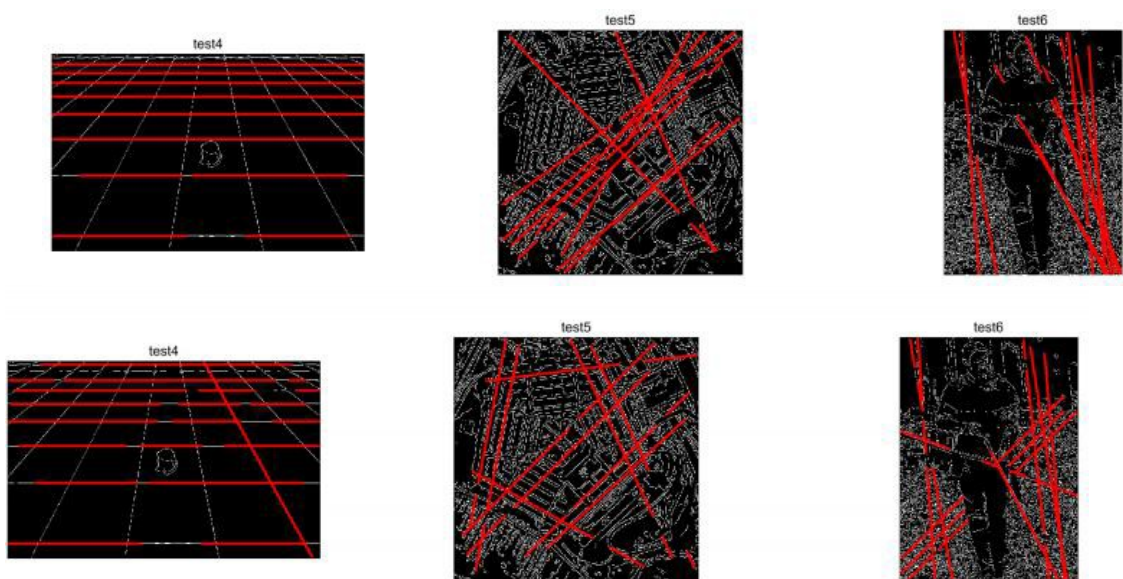
Sobel 边缘检测+Hough 变换，细分值为 1



Canny 边缘检测+Hough 变换，细分值为 1

不同 Hough 变换参数的结果

霍夫变换细分值为 0.7



霍夫变换细分值为 1

3 结果分析

3.1 比较不同边缘检测算法

hough 变换后经常会出现一些原图中不存在直线，但观察原图后发现其近似存在一条抽象的直线，Robert 算子得到直线相对较长，而 sobel 算子得到的直线相对较短。随着间隔的增大，canny 算子，sobel 算子，和 robert 算子都会得到一些抽象的直线。

4.程序代码

程序代码：

```
im1=imread('test1.tif');  
  
im2=imread('test2.png');  
  
im3=imread('test3.jpg');
```

```
g');
```

```
im4=imread('test4.bm
```

```
p');
```



```
im5=imread(' test5.  
  
png');  
  
im6=imread(' test6.  
  
jpg');  
  
im1=im1(:, :, 1);  
  
a=edge(im1,'cann  
  
y');  
  
b=edge(im2,'cann  
  
y');  
  
c=edge(im3,'cann  
  
y');  
  
d=edge(im4,'cann  
  
y');  
  
e=edge(im5,'cann  
  
y');  
  
f=edge(im6,'can  
  
ny');  
  
  
figure;  
  
subplot(2,3,1),  
  
imshow(a),axis  
  
on;  
  
title(' test1')
```

```
;

subplot(2,3,2),

imshow(b),axis on;

title(' test2' );

subplot(2,3,3),

imshow(c),axis on;

title(' test3' );

subplot(2,3,4),
```

```
imshow(d), axis on;
```

```
title(' test4' );
```

```
subplot(2,3,5),
```

```
imshow(e), axis on;
```

```
title(' test5' );
```

```
subplot(2,3,6),
```

```
imshow(f), axis on;
```

```
title(' test6' );
```

```
im=imread(' test2. png' );
```

```
a=edge(im, ' sobel' , 0.02);
```

```
b=edge(im, ' sobel' , 0.07);
```

```
c=edge(im, ' sobel' );
```

```
figure;
```

```
subplot(1,3,1),
```

```
imshow(a), axis
```

```
on;
```

```
title(' 阈值=0.02' );
```

```
subplot(1,3,2),
```

```
imshow(b), axis on;
```

```

title(' 阈值=0.07');

subplot(1,3,3

),

imshow(c), axis

on;

title(' 默认');


BW=imread(' test2. png');

%BW=rgb2gray(BW

);

thresh=[0.01,0

.17];

sigma=2;%定义高斯参数

%f =

edge(double(BW),'canny',thresh,sig

ma); f =

edge(double(BW),'canny');

figure(1),imshow(f,[]);

title('canny 边缘检测');


[H, theta, rho]=hough(f,'RhoResolution',0.5);

%imshow(theta,rho,H,[],'notruesize'),axis on,axis normal

```

```
%xlabel(' \theta'), ylabel(' rho' );
```

```
peak=houghpeaks(H, 5);
```

hold on

```
lines=houghlines(f, theta, rho, peak);
```

```
figure, imshow(f, []), title('Hough Transform
```

```
Detect Result'), hold on for k=1:length(lines)
```

```
    xy=[lines(k).point1;lines(k).poin
```

```
    t2];
```

```
    plot(xy(:,1), xy(:,2), 'LineWidth', 4,
```

```
        'Color', [.6 .6 .6]);
```

```
end
```

```
im1=imread('test1.t
```

```
if');
```

```
im2=imread('test2.pn
```

```
g');
```

```
im3=imread('test3.j
```

```
pg');
```

```
im4=imread('test4.bm
```

```
p');
```

```
im5=imread('test5.pn
```

```
g');
```

```
im6=imread('test6.j
```

```
pg');
```

```
im1=im1(:, :, 1);
```

```
a=edge(im1,'sob
```

```
el');
```

```
b=edge(im2,'sob
```

```
el');
```

```
c=edge(im3,'sob
```

```
el');
```

```
d=edge(im4,'sob
```

```
el');
```

```
e=edge(im5,'sob
```

```
el');
```

```
f=edge(im6,'sob
```

```
el');
```

```
figure;

subplot(2,3,

1),

imshow(a),ax

is on;

title('test

1');

subplot(2,3,2),

imshow(b),axis

on;

title('test2');

subplot(2,3,3),

imshow(c),axis

on;

title('test3');

subplot(2,3,4),

imshow(d),axis

on;

title('test4');

subplot(2,3,5),
```



```
imshow(e), axis  
  
on;  
  
title(' test5');  
  
subplot(2  
  
, 3, 6),  
  
imshow(f),  
  
axis on;  
  
title(' te  
  
st6')
```