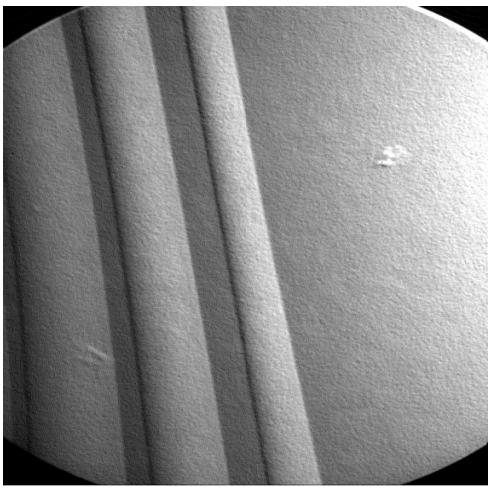
直线检测实验报告

杨德宇 自动化 66 2161500050

一．边缘检测

测试图像 test1~test6 的原图像如下所示：

test1

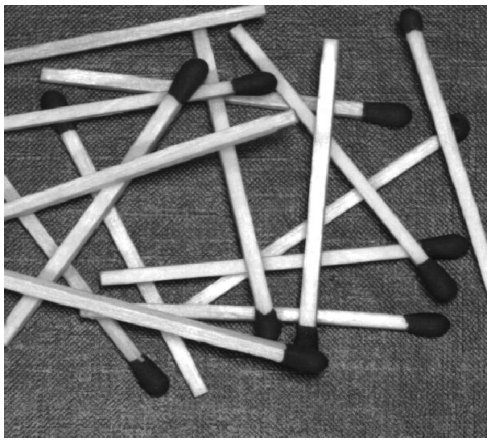


test2

test3

test4

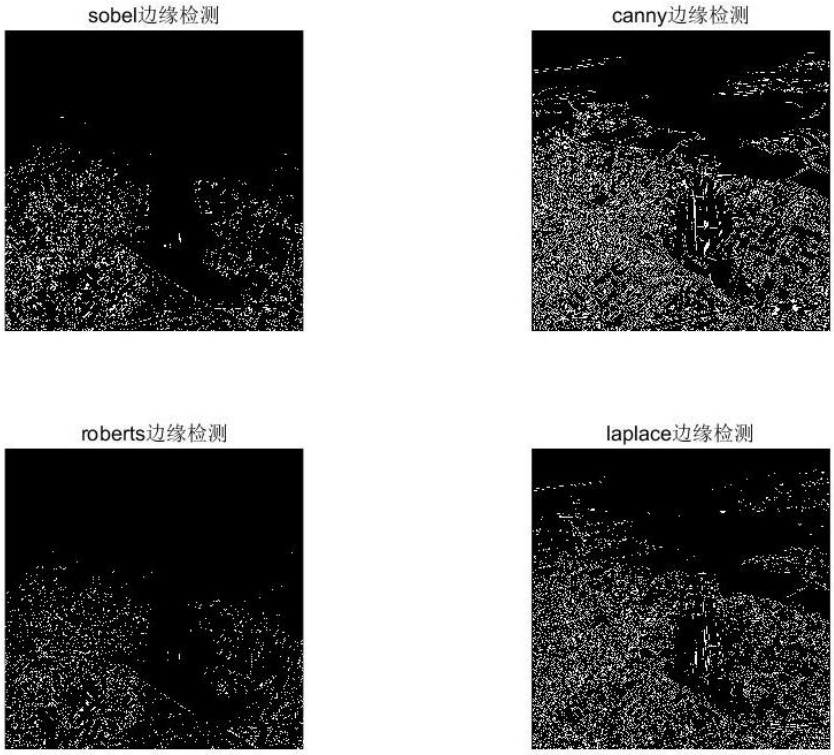
test5



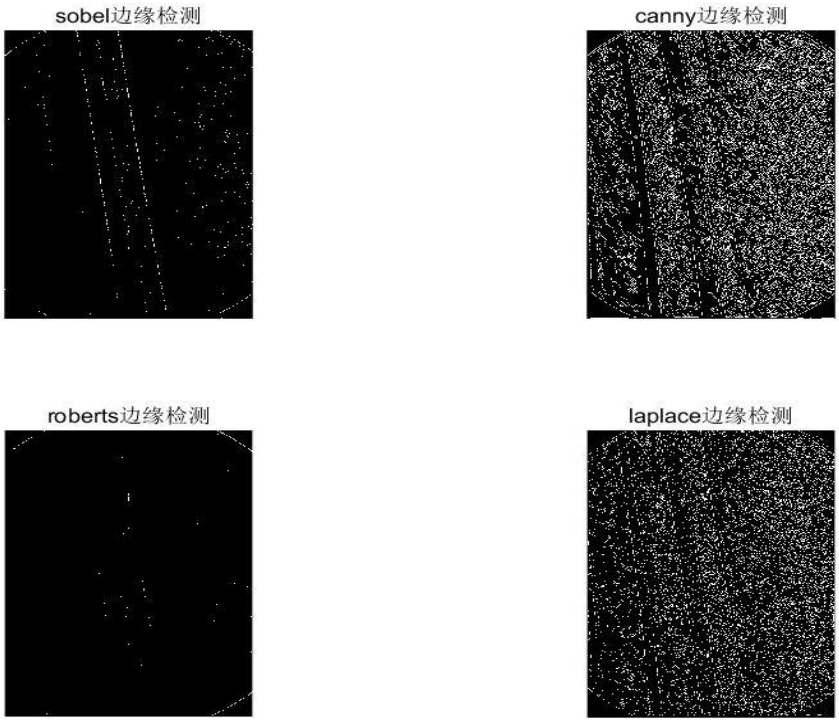
test6

分别采用 Sobel 算子，Canny 算子，Roberts 算子和 Laplace 算子对 6 幅图像

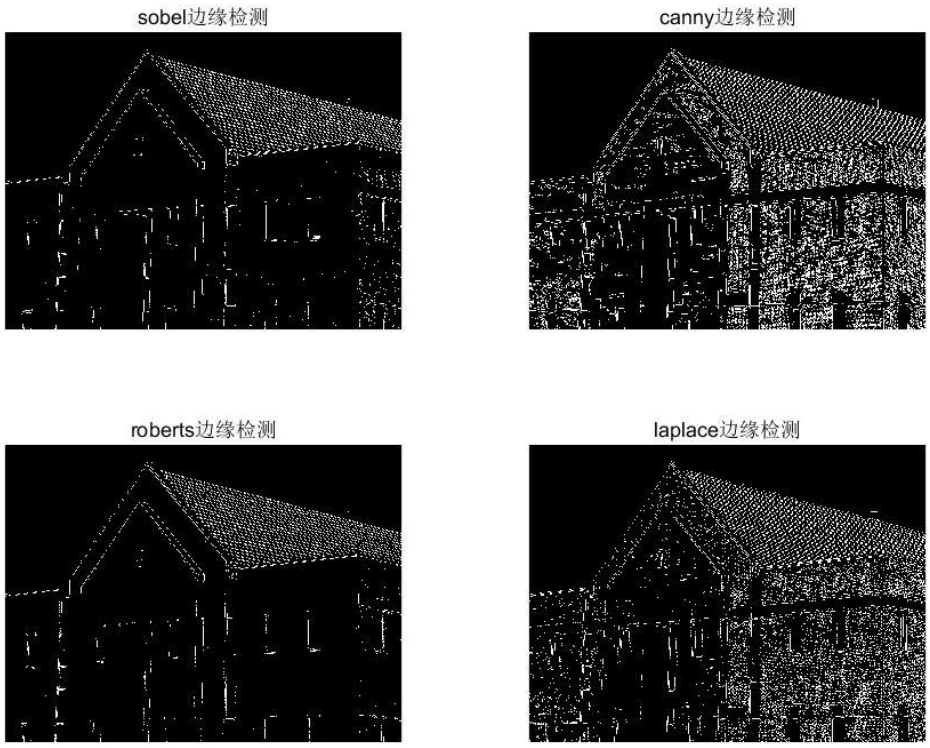
进行边缘检测。MATLAB 2016a 中提供的边缘检测函数 edge，调用格式为 I=edge(image,type),其中 type 可指定采用的算子类型。实现的效果如下：



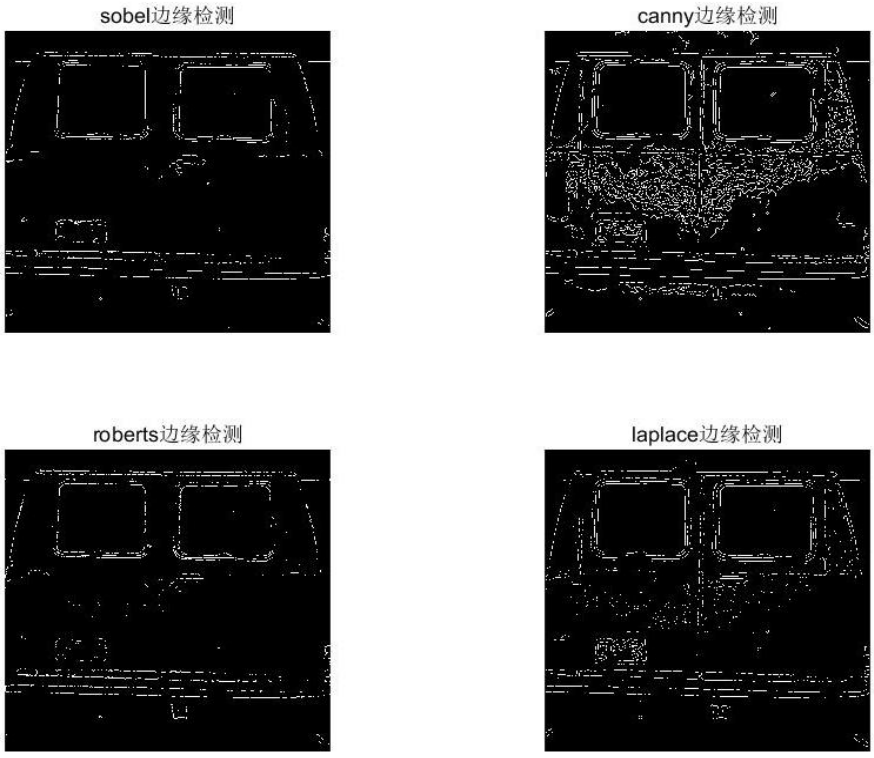
test1 边缘检测



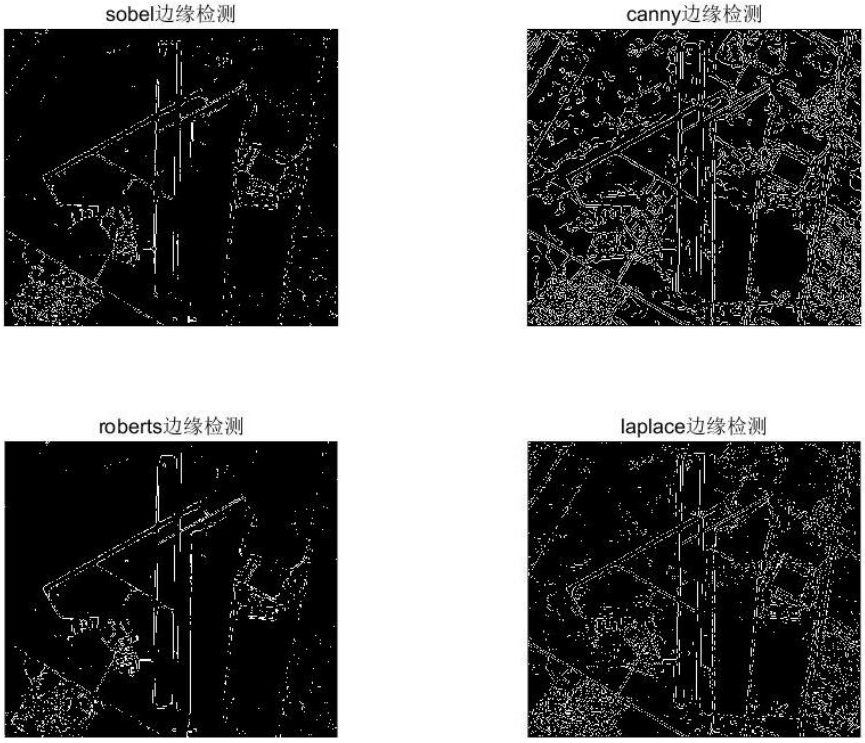
test2 边缘检测



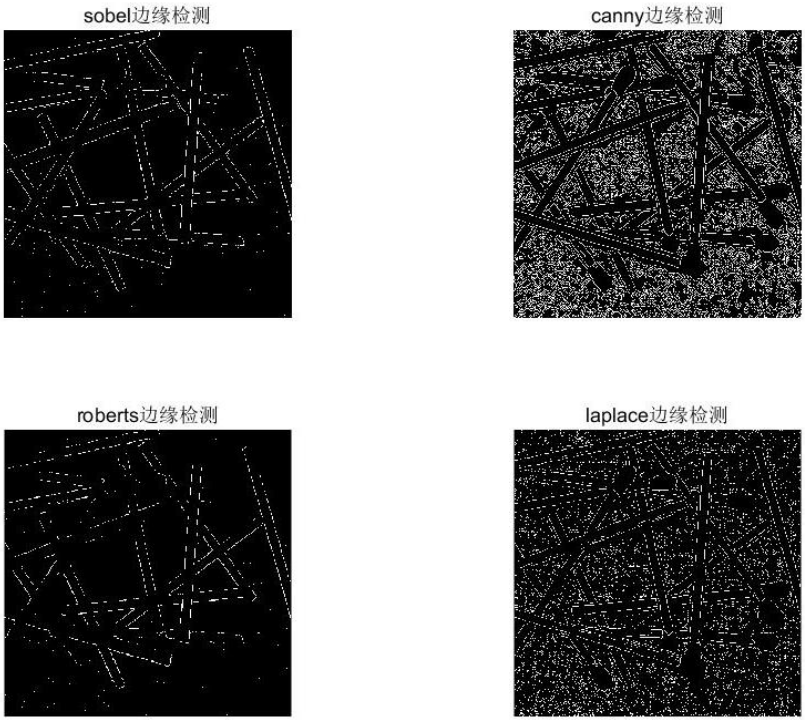
test3 边缘检测



test4 边缘检测



test5 边缘检测



test6 边缘检测

从边缘检测的效果来看，采用 canny 算子和 laplace 算子的检测方法明显比其

他方法要多出更多的细节部分，而 sobel 算子和 roberts 算子只能检测出变化明显的部分。并且由于 roberts 算子利用的是局部差分，边缘部分容易产生模糊，在图中也有较为明显的体现。

二．Hough 直线检测

2.1 直线检测原理

霍夫变换(Hough Transform)是图像处理中的一种特征提取技术，可以识别图像中的几何形状。它将图像空间中的特征点映射到参数空间进行投票，通过检测累计结果的局部极值点得到一个符合某特定形状的点的集合。经典霍夫变换用来检测图像中的直线。

其算法步骤如下：

1、得到图像的边缘信息；

2、对边缘图像中的每一个点，在 k-b 空间中画出一条直线；

3、对各直线上的点，我们采取“投票”（vote）的方法，即累加：有直线经过这一点，这一点的值加 1；

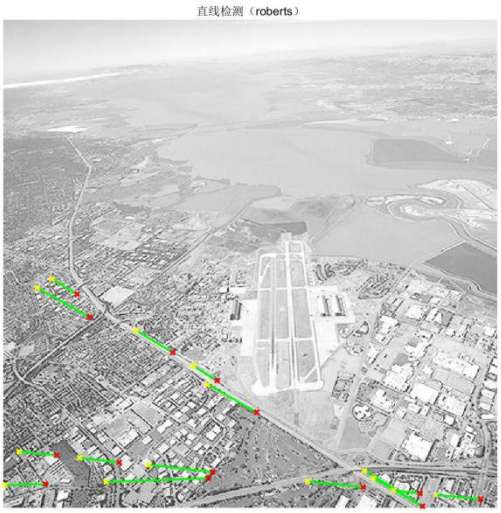
4、遍历 k-b 空间，找出局部极大值点，这些点的坐标（k，b）就是原图像中可能的直线的斜率和截距。

2.2 检测效果

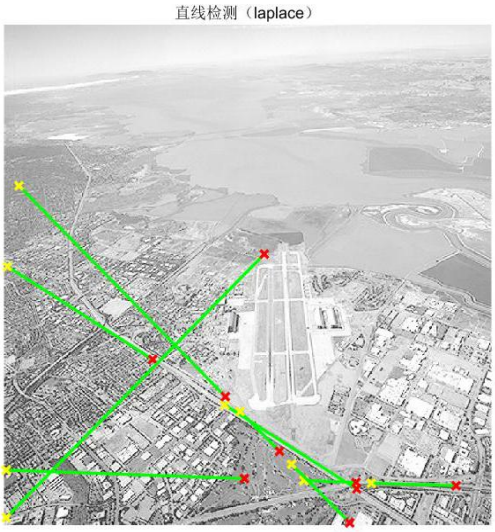
参数选取 4 个极值点时，效果如下：

、

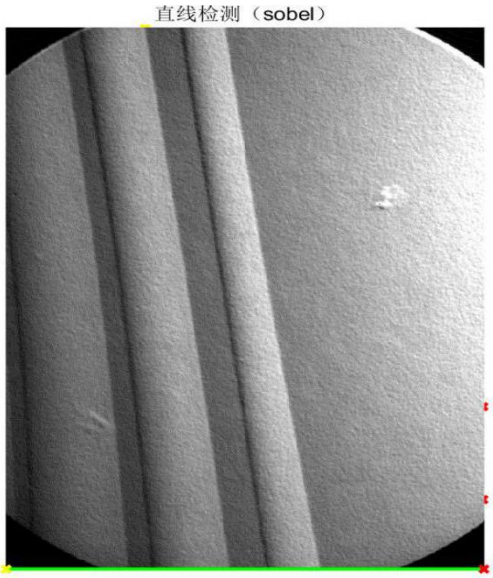
test1-sobel　　**test1-canny**



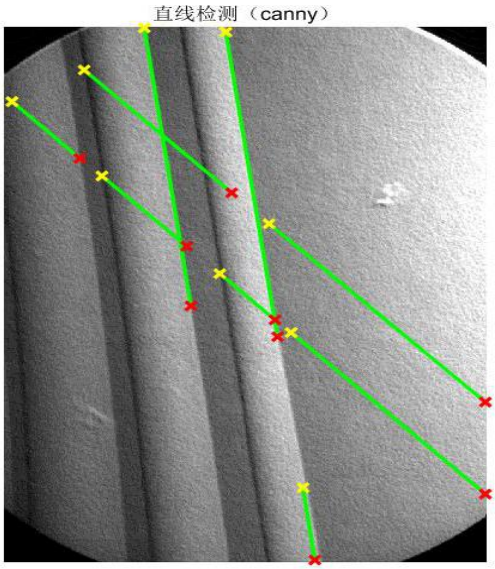
test1-roberts



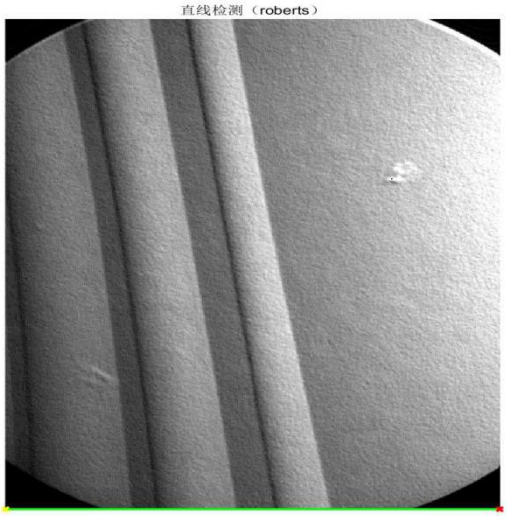
test1-laplace



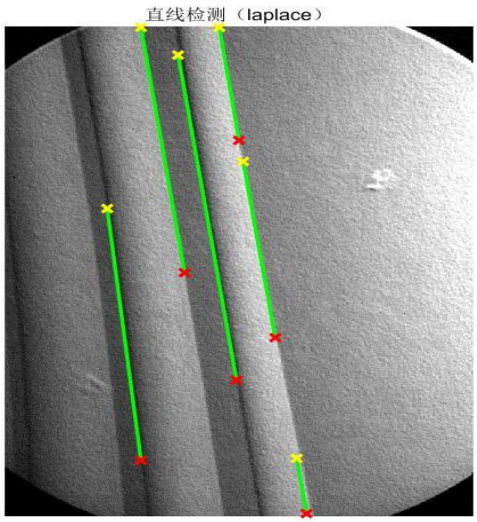
test2-sobel



test2-canny

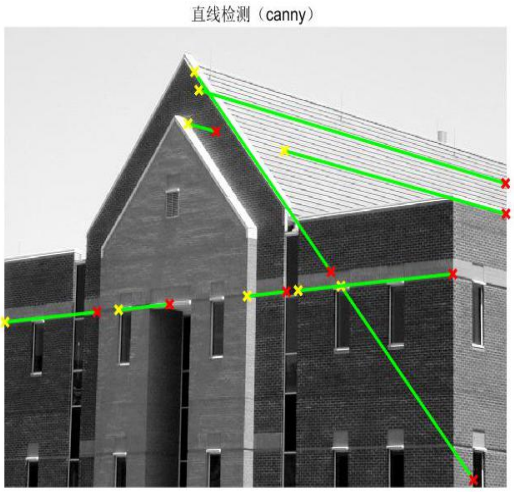


test2-roberts

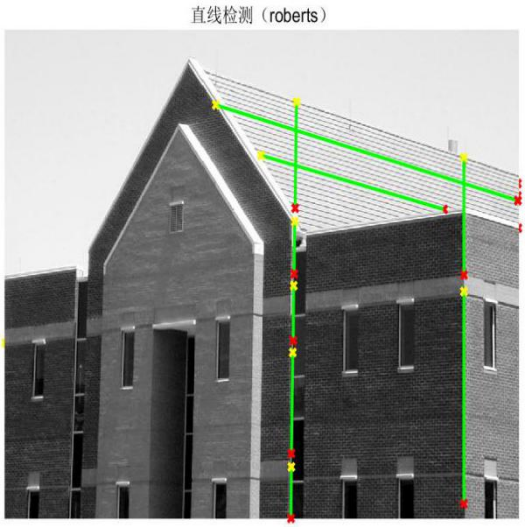


test2-laplace

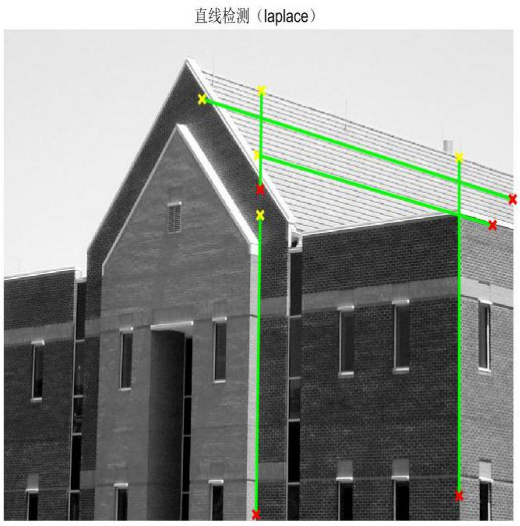
test3-sobel



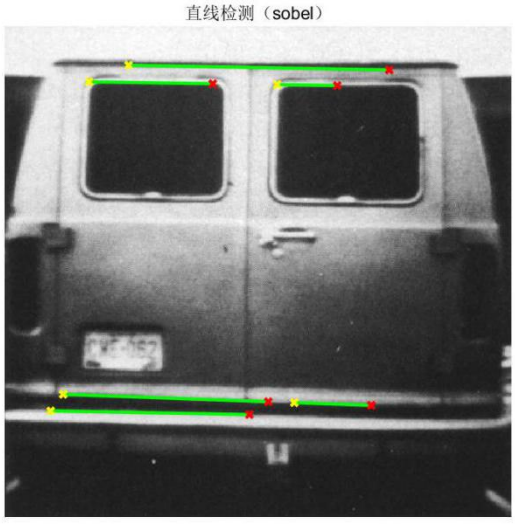
test3-canny



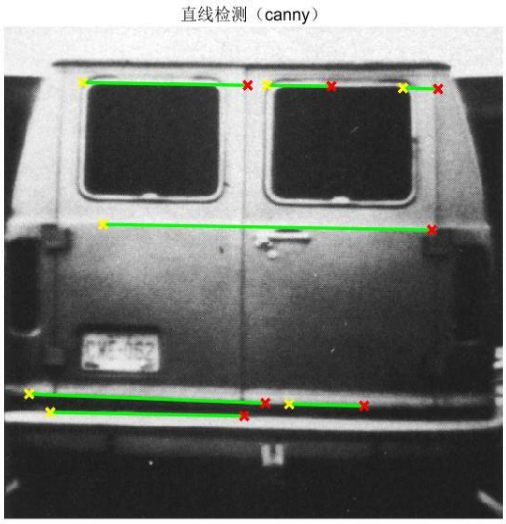
test3-roberts



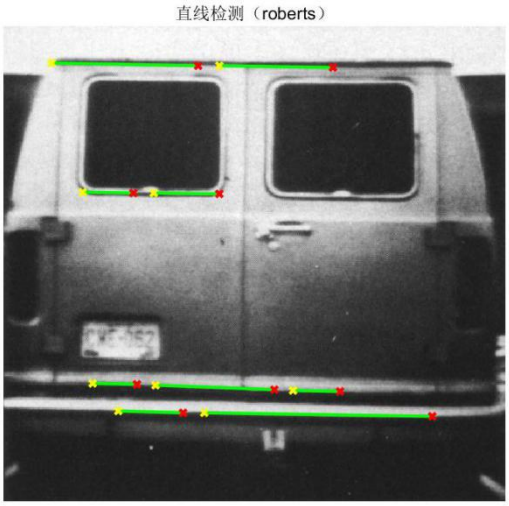
test3-canny



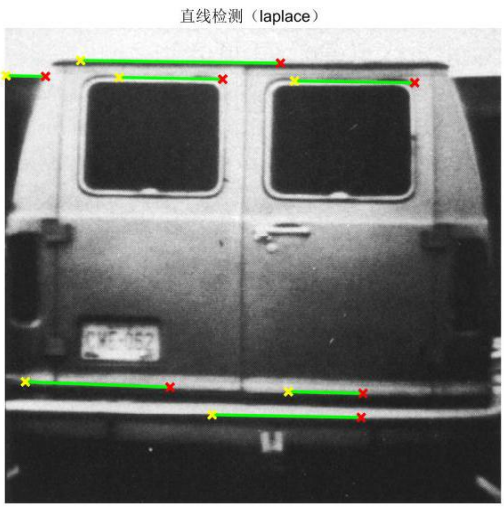
test4-sobel



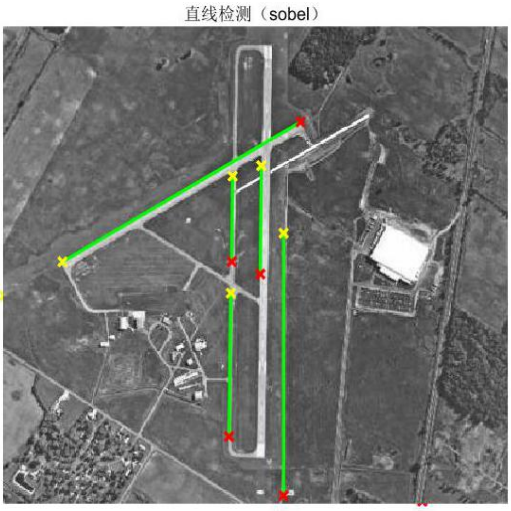
test4-canny



test4-roberts

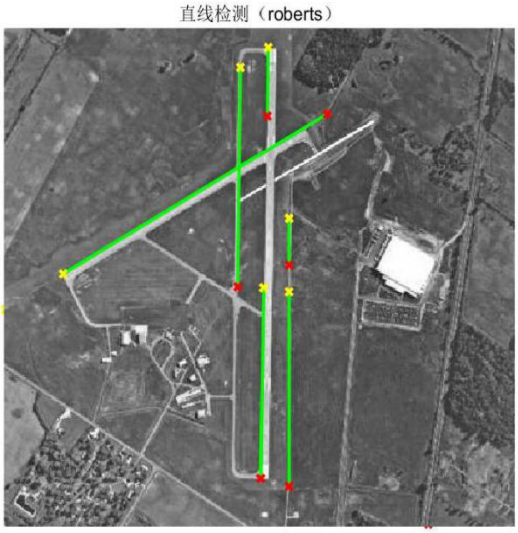


test4-laplace

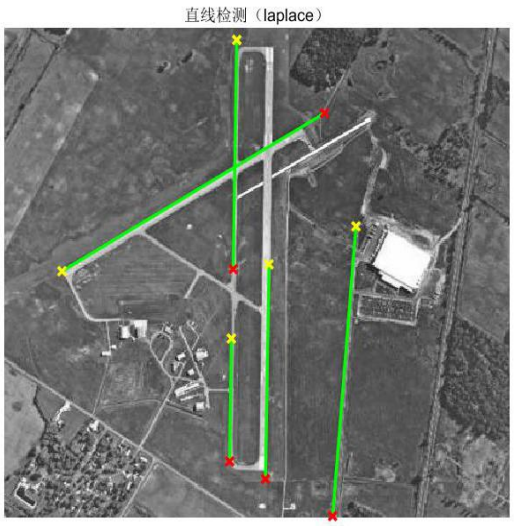


test5-sobel

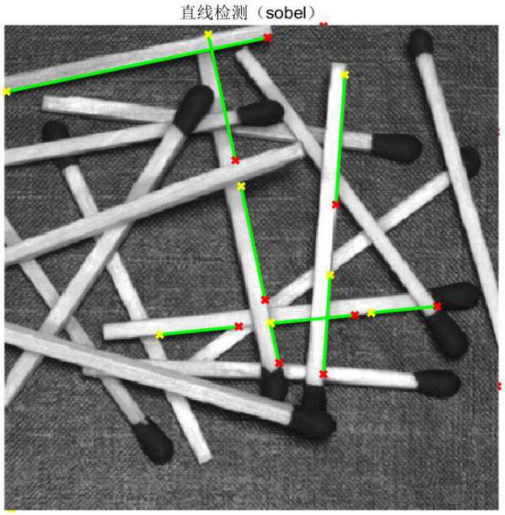
test5-canny



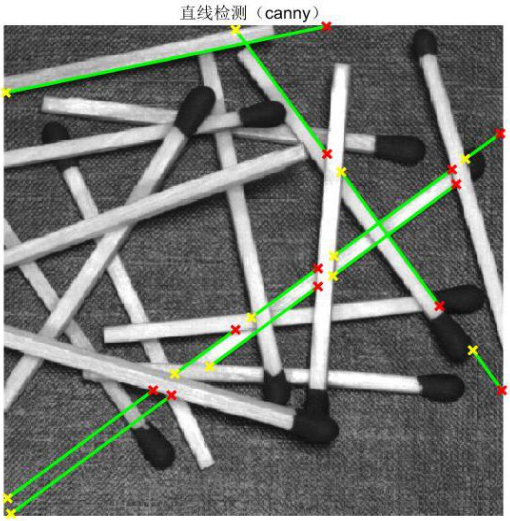
test5-roberts



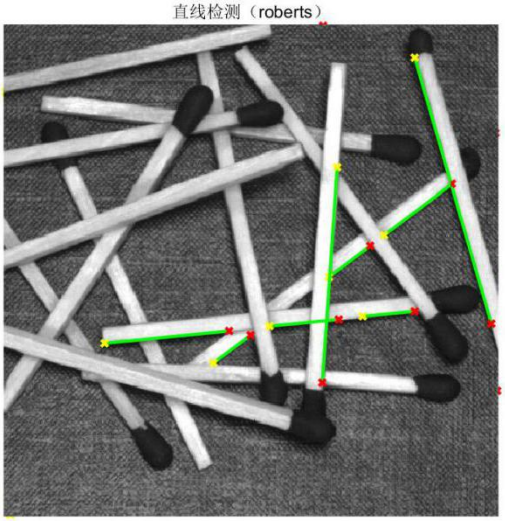
test5-laplace



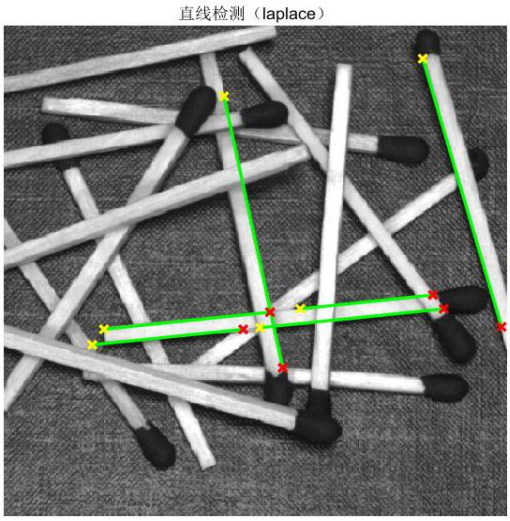
test6-sobel



test6-canny



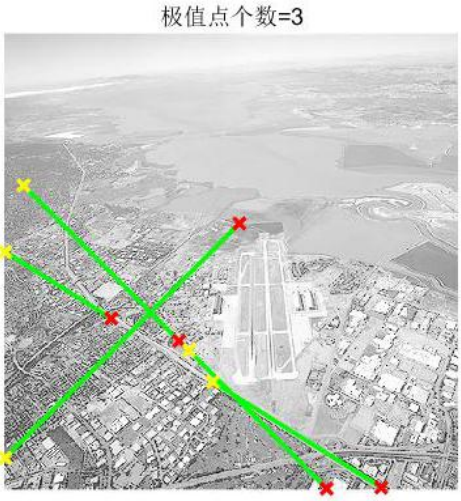
test6-roberts

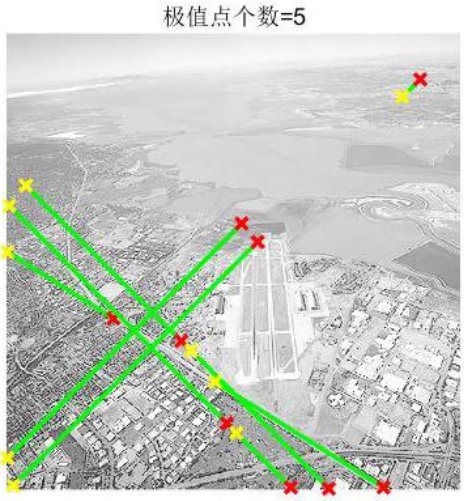


test6-laplace

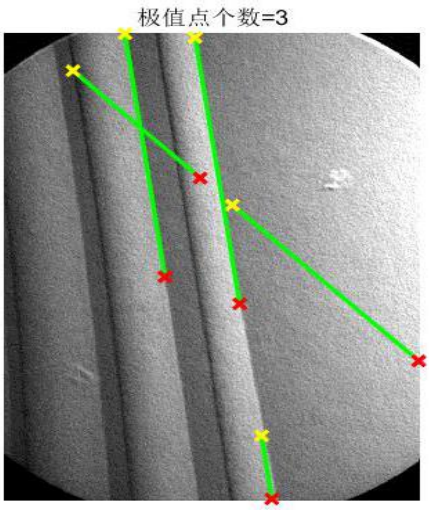
从 6 幅图像的直线检测结果来看，经过 Canny 边缘检测后的图像明显能检测出更多的直线，检测效果明显优于其它几种边缘检测算法。对于 test2 图像，采用 Sobel 和 Roberts 边缘检测算子未能检测出图像中的直线，效果较差；拉普拉斯边缘检测算子表现比 Caany 算子稍差，但优于 Sobel 和 Roberts 算子。可以发现的是，不管是哪种边缘检测算法，处理后利用霍夫变换都无法检测出人眼可以观察到的全部直线，这也正说明人眼视觉系统和计算机视觉的差别。在人眼看起来很明显的东西，在计算机中可能就不那么容易了。

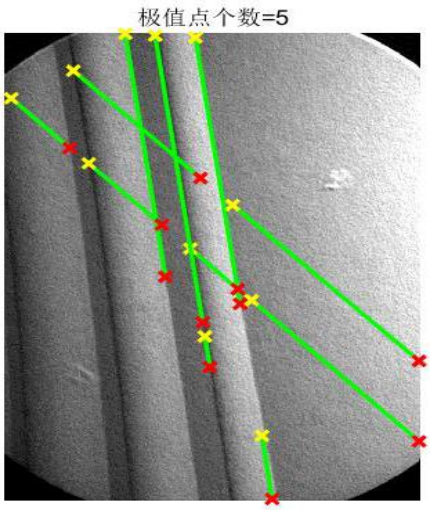
以上检测结果是基于 4 个极值点，若改变极值点个数，采用 Canny 算子进行边缘检测，得到的效果如下。



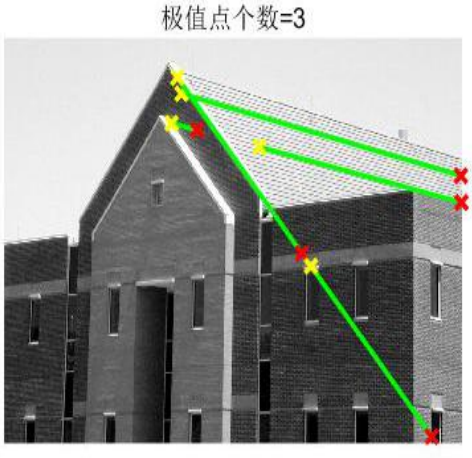


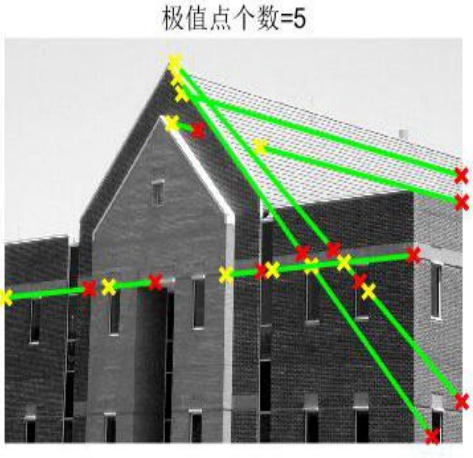
Test1

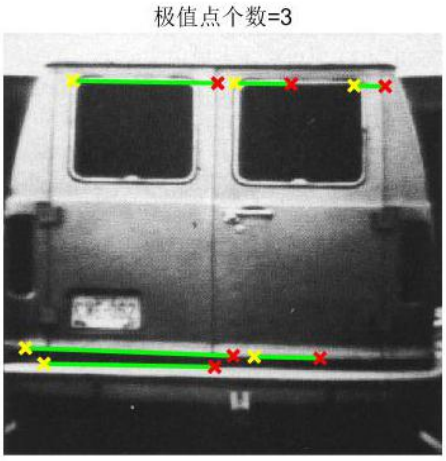


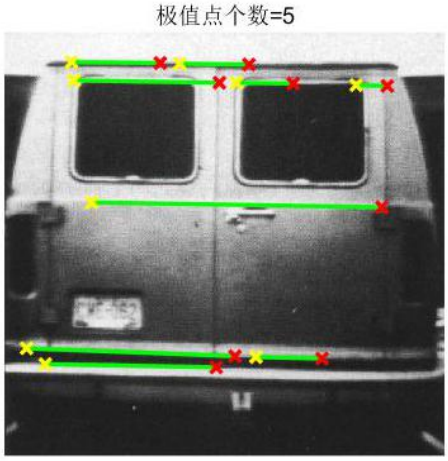


Test2

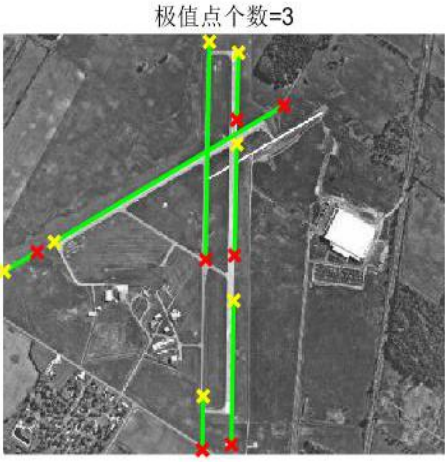


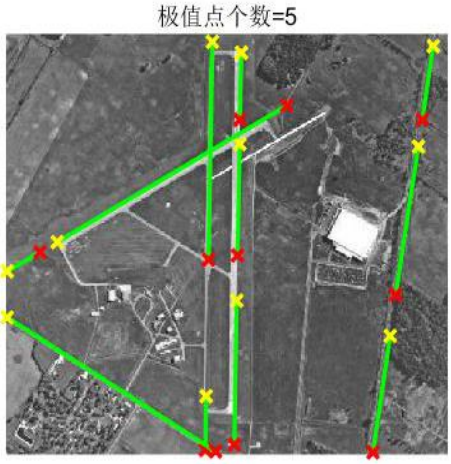




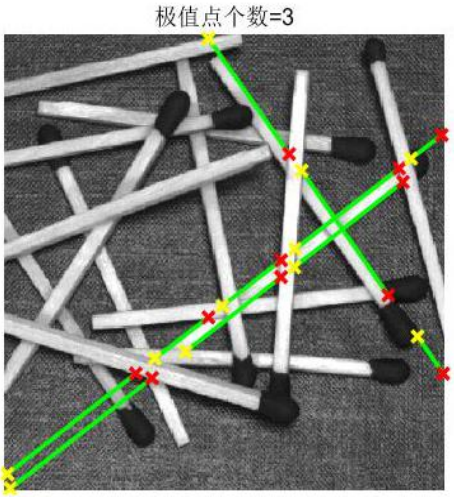


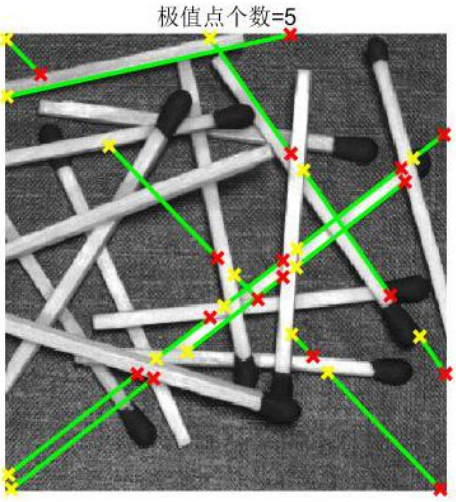
Test4





Test5





Test6

从几幅图像的处理结果来看，极值点个数增加明显增加了检测出的直线条数，但这也容易带来一些“假直线”的检测结果。极值点参数的选择依赖于图片的具体情况和检测人的意图，如果希望检测出更多的直线则可以将极值点个数增加。

附录

1. 边缘检测

```
clc;clear;
D=dir('E:\1111\数字图像处理\第七次作业\*.tif')% 构成文件路径和名称
for i=1:length(D)
    f=imread(['E:\1111\数字图像处理\第七次作业\' D(i).name]);
    sobel_image=edge(f,'sobel');
    canny_image=edge(f,'canny');
    roberts_image=edge(f,'roberts');
    laplc_image=edge(f,'log');
    figure(i);
    subplot(2,2,1);imshow(sobel_image);title('sobel 边缘检测');
    subplot(2,2,2);imshow(canny_image);title('canny 边缘检测');
    subplot(2,2,3);imshow(roberts_image);title('roberts 边缘检测');
    subplot(2,2,4);imshow(laplc_image);title('laplace 边缘检测');
    saveas(i,strcat(D(i).name(1:5),'边缘检测'),'jpg');
end
```

2. 直线检测

极值点个数为 3

```
clc;clear;
D=dir('E:\1111\数字图像处理\第七次作业\*.tif')% 构成文件路径和名称
for i=1:length(D)
    I=imread(['E:\1111\数字图像处理\第七次作业\' D(i).name]);
%       bw=edge(I,'sobel');
%       bw=edge(I,'canny');
%       bw=edge(I,'roberts');
    bw=edge(I,'log');
    [H,T,R]=hough(bw);%计算二值图像的标准霍夫变换，H 为霍夫变换矩阵，I,R 为计算霍夫变换的角度和半径值
    % imshow(H,[],'XData',T,'YData',R,'InitialMagnification','fit');
    P=houghpeaks(H,4);
    x1=T(P(:,2));
    y1=R(P(:,1));
%   plot(x,y,'s','color','blue');%标出极值点
    lines=houghlines(bw,T,R,P);%提取线段
    figure(i);
    imshow(I);hold on;
```

```
    for k = 1:length(lines)
        xy = [lines(k).point1; lines(k).point2];
        plot(xy(:,1),xy(:,2),'LineWidth',2,'Color','green');%画出线段
        plot(xy(1,1),xy(1,2),'x','LineWidth',2,'Color','yellow');%起点
        plot(xy(2,1),xy(2,2),'x','LineWidth',2,'Color','red');%终点
    end
%       title('直线检测（sobel）');
%       saveas(i,strcat(D(i).name(1:5),'_直线检测(sobel)'),'jpg');
%       title('直线检测（canny）');
%       saveas(i,strcat(D(i).name(1:5),'_直线检测(canny)'),'jpg');
%         title('直线检测（roberts）');
%         saveas(i,strcat(D(i).name(1:5),'_直线检测(roberts)'),'jpg');
      title('直线检测（laplace）');
      saveas(i,strcat(D(i).name(1:5),'_直线检测(laplace)'),'jpg');
end
```

不同极值点个数

```
clc;clear;
D=dir('E:\1111\数字图像处理\第七次作业\*.tif')% 构成文件路径和名称
for i=1:length(D)
    I=imread(['E:\1111\数字图像处理\第七次作业\' D(i).name]);
    bw=edge(I,'canny');
    [H,T,R]=hough(bw);%计算二值图像的标准霍夫变换，H 为霍夫变换矩阵，I,R
为计算霍夫变换的角度和半径值
    % imshow(H,[],'XData',T,'YData',R,'InitialMagnification','fit');
    P1=houghpeaks(H,3); P2=houghpeaks(H,5);
    x1=T(P1(:,2));x2=T(P2(:,2));
    y1=R(P1(:,1));y2=R(P2(:,1));
    lines1=houghlines(bw,T,R,P1);%提取线段
    lines2=houghlines(bw,T,R,P2);%提取线段
    figure(i);
    subplot(1,2,1);imshow(I);title('极值点个数=3');hold on;
    for k = 1:length(lines1)
        xy = [lines1(k).point1; lines1(k).point2];
        plot(xy(:,1),xy(:,2),'LineWidth',2,'Color','green');%画出线段
        plot(xy(1,1),xy(1,2),'x','LineWidth',2,'Color','yellow');%起点
        plot(xy(2,1),xy(2,2),'x','LineWidth',2,'Color','red');%终点
    end
    subplot(1,2,2);imshow(I);title('极值点个数=5');hold on;
    for k = 1:length(lines2)
        xy = [lines2(k).point1; lines2(k).point2];
        plot(xy(:,1),xy(:,2),'LineWidth',2,'Color','green');%画出线段
        plot(xy(1,1),xy(1,2),'x','LineWidth',2,'Color','yellow');%起点
        plot(xy(2,1),xy(2,2),'x','LineWidth',2,'Color','red');%终点
```

```
        end
            saveas(i,strcat(D(i).name(1:5),'_极值点'),'jpg');
end
```