数字图像处理第七次作业

姓名：边策

班级：自动化 63

学号：2160504062

摘要：本次作业是基于sobel算子，prewitt算子和canny双阈值算法对六幅图像进行边缘检测。并通过对不同的边缘检测方法，利用哈夫变换进行全局边缘提取，比较不同。

报告正文

一、实验原理

sobel 算子

根据所学，我们可知，sobel 算子是一种微分的算子，在当检测垂直与水平方向的直线时，掩模形式如下图。

索贝尔算子（Sobel operator）主要用作边缘检测，在技术上，它是一离散性差分算子，用来运算图像亮度函数的灰度之近似值。在图像的任何一点使用此算子，将会产生对应的灰度矢量或是其法矢量

Sobel卷积因子为：

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

该算子包含两组3x3的矩阵，分别为横向及纵向，将之与图像作平面卷积，即可分别得出横向及纵向的亮度差分近似值。如果以A代表原始图像，Gx及Gy分别代表经横向及纵向边缘检测的图像灰度值，其公式如下：

$$\mathbf{G_x} = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * \mathbf{A} \quad \text{and} \quad \mathbf{G_y} = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * \mathbf{A}$$

如果想关心对角线方向的直线，我们还有别的掩模形式。在书中可以找到。通过进行空域卷积得到$g_x(\mathrm{x},\mathrm{y})$，$g_y(\mathrm{x},\mathrm{y})$。并用下面公式得到检测的图像

$$\mathrm{M}(\mathrm{x},\mathrm{y}) = \sqrt{{g_x}^2 + {g_y}^2}$$

2. prewitt 算子

实现原理和 sobel 算子一样，只不过其算子中间部分都为 1。两者本身没有太多的区别，但是 sobel 算子能够平滑噪声，所以实现起来 sobel 算子比 prewitt 算子效果会更好一些。

3. Canny 边缘检测原理

Canny 边缘检测算法可以分为以下 5 个步骤：

1) 使用高斯滤波器，以平滑图像，滤除噪声。 2) 计算图像中每个像素点的梯度强度和方向。
3) 应用非极大值（Non-Maximum Suppression）抑制，以消除边缘检测带来的杂散响应。
4) 应用双阈值（Double-Threshold）检测来确定真实的和潜在的边缘。
5) 通过抑制孤立的弱边缘最终完成边缘检测。

4.哈夫变换

哈夫变换方法是利用图像得全局特性而对目标轮廓进行直接检测的方法，在已知区域形状的条件下，哈夫变换可以准确地捕获到目标的边界（连续的获不连续的），并最终以连续曲线的形式输出变换结果，该变换可以从强噪声环境中将已知形状的目标准确得分割提取出来。

哈夫变换的核心思想是: 点—线的对偶性。通过变换将图象从图像控件转换到参数空间,在图像空间中一条过点(x,y)的直线方程为 y=ax+b,通过代数变换可以转换为另一种形式 b=-ax+y,即参数空间中过点(a,b)的一条直线,如果在图像空间中保持直线的斜率和截距的不变,其在参数空间必定过点(a,b),这也就说明,在图像空间中共线的点对应参数空间共点的线。哈夫变换就是根据上述点—线的对偶性把在图象空间中存在的直线检测问题转换为参数空间中存在的点检测问题,后者的处理要比前者简单易行得多,只需简单地累加统计即可实现对边缘的检测。

直线到直线的参数方程存在一个问题，如果直线接近竖直方向，会由于 a 和 b 的值都接近无穷而使计算量大增。因此实际中大量采用的是直线的极坐标方程：

$$\rho = xcos\theta + ysin\theta$$

根据这个方程，原图像空间中的点对应新参数空间中的一条正弦曲线，即原来的点—直线对偶性变成了现在的点—正弦曲线对偶性。检测在图像空间中共点的线需要在参数空间里检测正弦曲线的交点。具体的检测步骤如下：

1. 在参数空间 ρθ 里建立一个两维的累加数组——哈夫矩阵。设这个累加数组为 H（ρ，θ）；
2. 开始时置数组 H 为零，然后对每一个图像空间中的给定点（x，y），让 θ 取遍 θ 轴上所有可能的值，并算出对应的 ρ（x，y 固定）；
3. 再根据 ρ 和 θ 的值（设都已经取整）对 H 累加：H（ρ，θ） = H（ρ，θ）+1；
4. 累加结束后，根据 H（ρ，θ）处共线点的个数。同时（ρ，θ）值也给出了参数方程，使我们得到了点所在三角函数曲线的方程；
5. 如果两个点（x1,y1)和(x2,y2)共线，则有相同的 ρ，θ，H（ρ，θ）则可以不断累加；
6. 根据哈夫矩阵 H 选择值最大的 N 个点，得到 xy 平面上的 N 条直线。

二、所用 matlab 函数

1.edge(img1,module);module可取'sobel', 'prewitt', 'canny'等值

输出值为边缘检测图像

2. [H,T,R] = hough(img)

计算二值图像的标准霍夫变换，H为霍夫变换矩阵，T,R为计算霍夫变换的角度和半径值

3. P = houghpeaks(H,jiaodian)

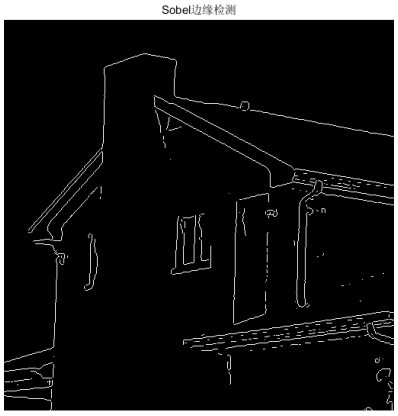
提取极值点

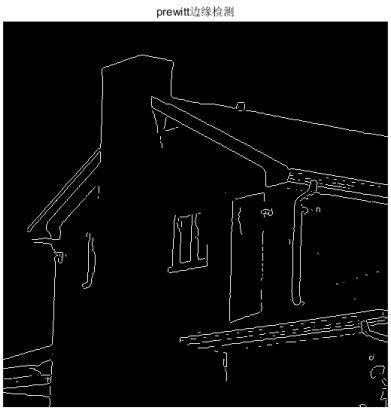
4. lines=houghlines(jiance,T,R,P)

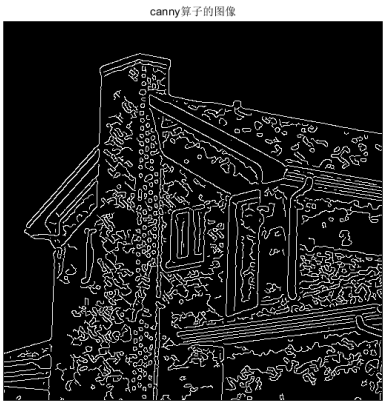
提取线段

三、实验结果

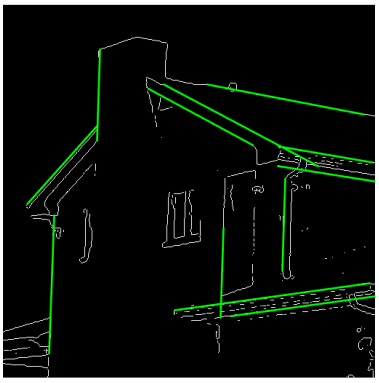
test1

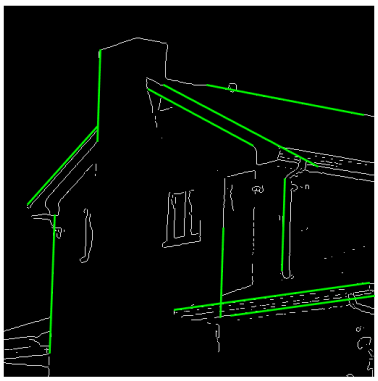




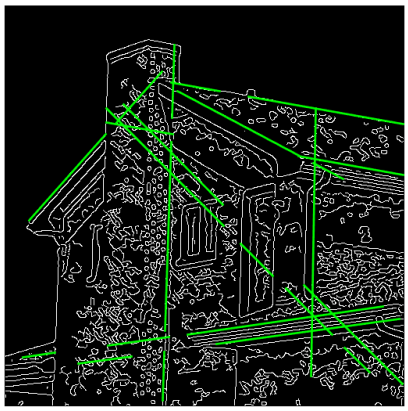


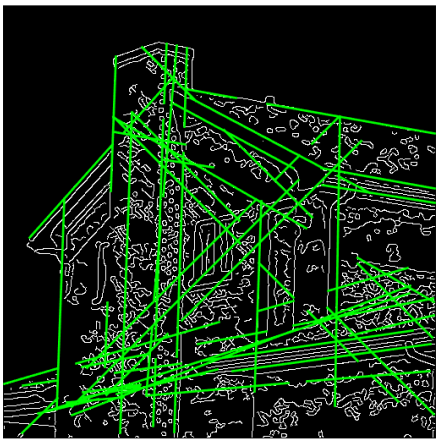
当极点数为 10 和 30 时，对 sobel 结果边缘提取



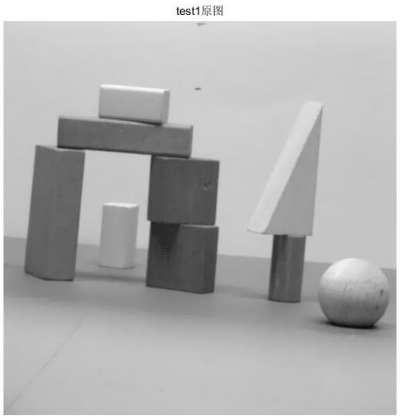


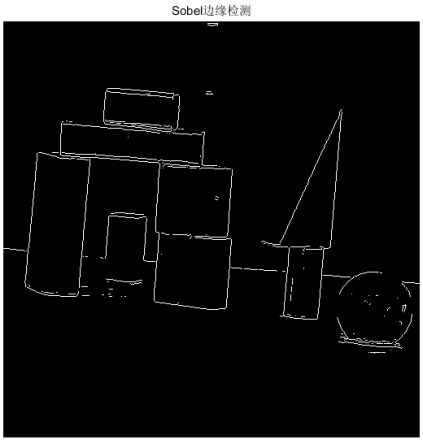
当极点数为 10 和 30 时，对 canny 结果边缘提取

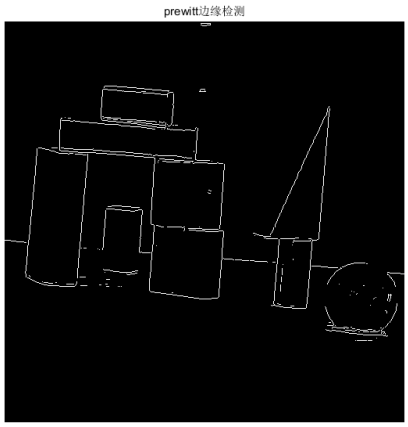


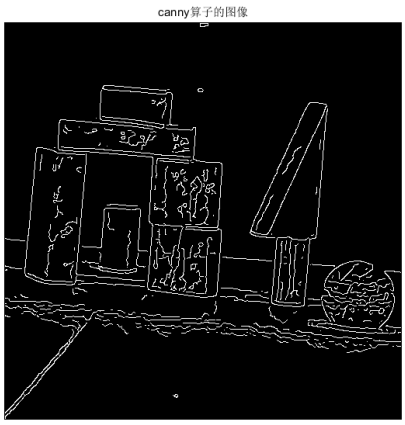


test2

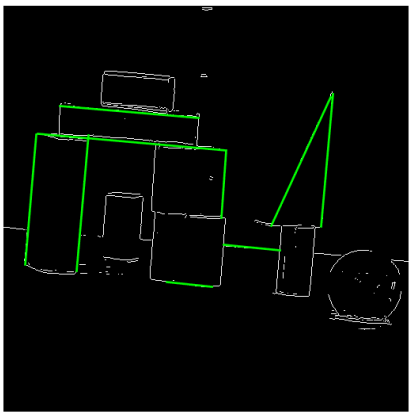


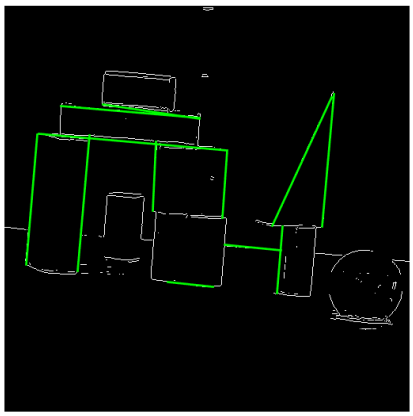




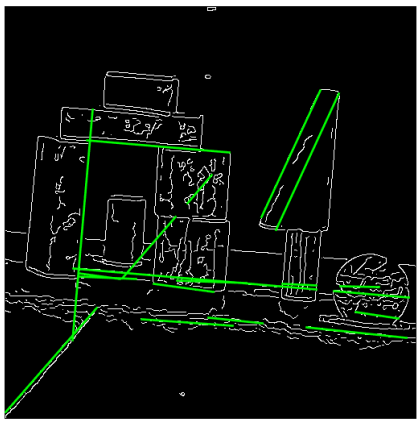


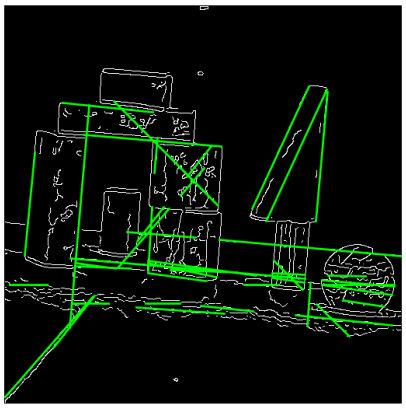
当极点数为 10 和 30 时，对 sobel 结果边缘提取



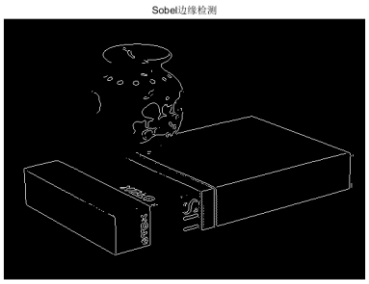


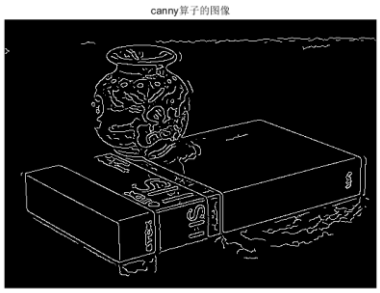
当极点数为 10 和 30 时，对 canny 结果边缘提取



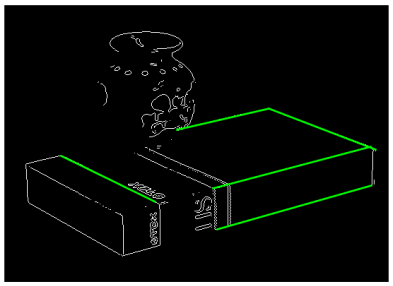


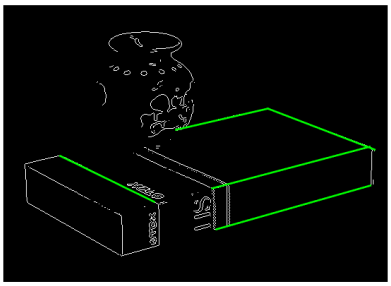
test3



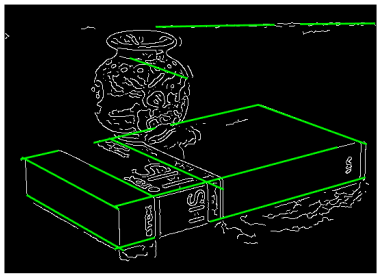


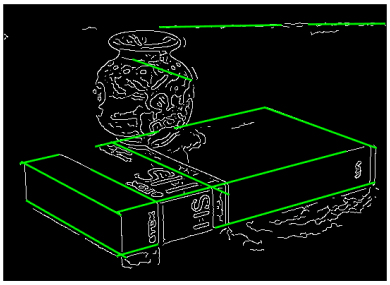
当极点数为 10 和 30 时，对 sobel 结果边缘提取



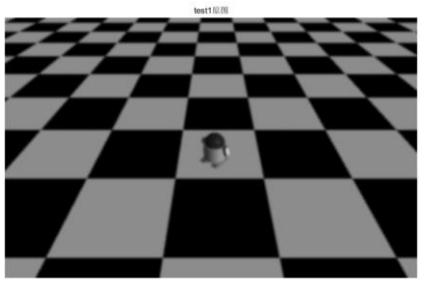


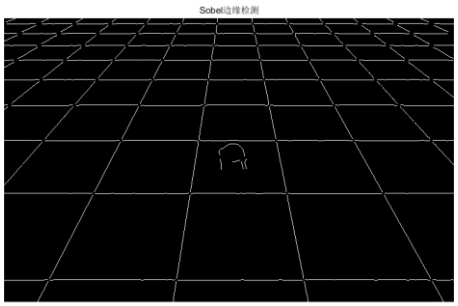
当极点数为 10 和 30 时，对 canny 结果边缘提取

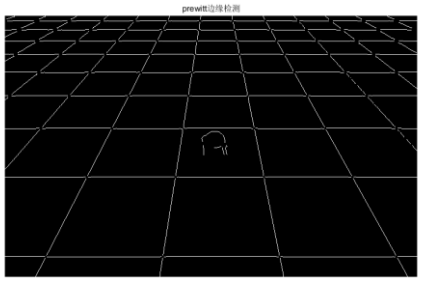


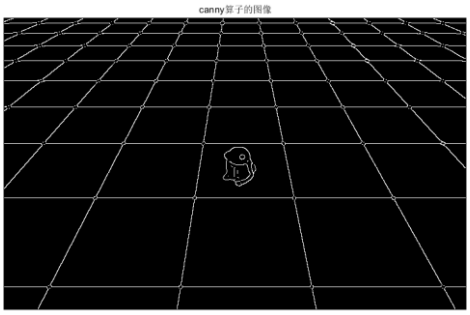


Test4

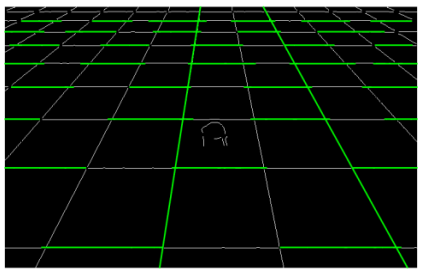


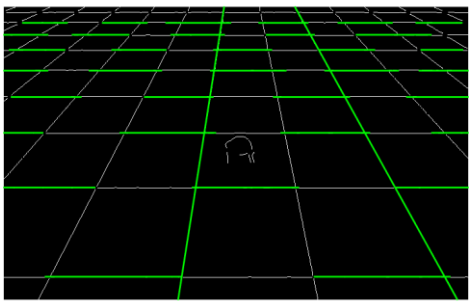




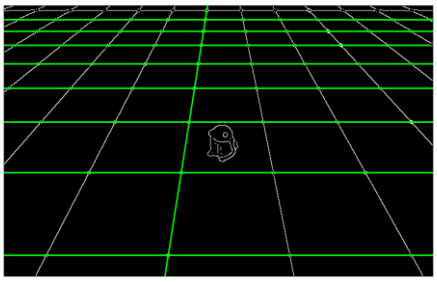


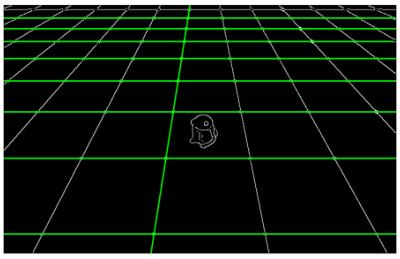
当极点数为 10 和 30 时，对 sobel 结果边缘提取



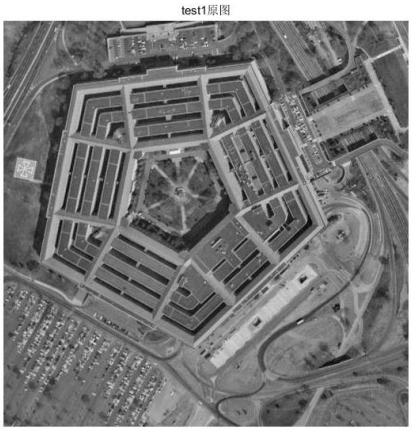


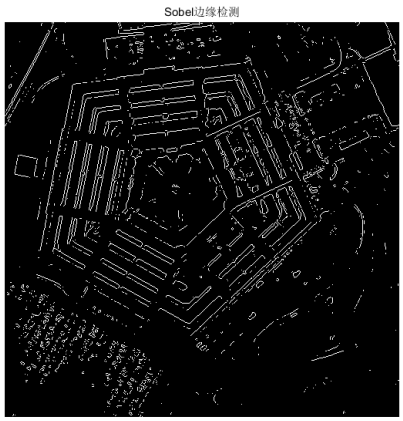
当极点数为 10 和 30 时，对 canny 结果边缘提取



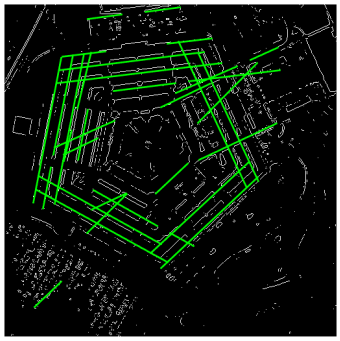


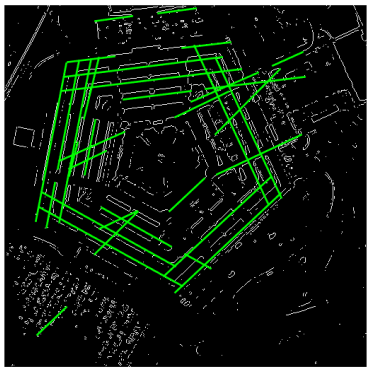
Test5



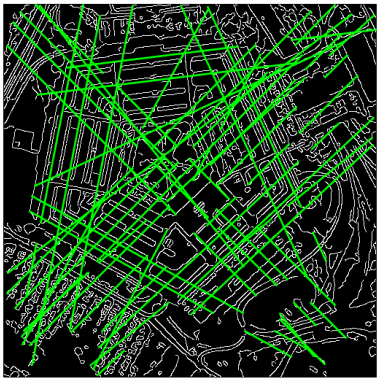


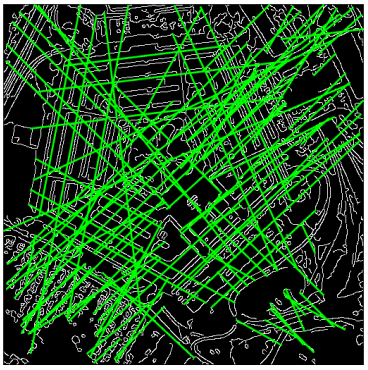
当极点数为 30 和 50 时，对 sobel 结果边缘提取





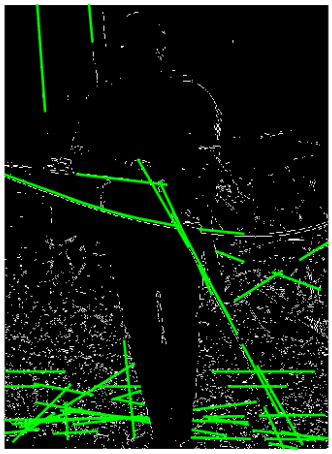
当极点数为 30 和 50 时，对 canny 结果边缘提取

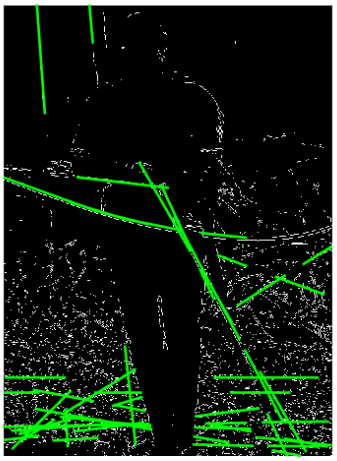




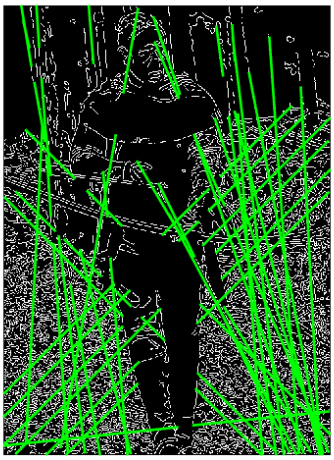
Test6

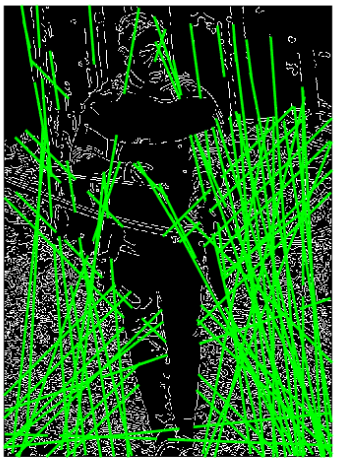
当极点数为 30 和 50 时，对 sobel 结果边缘提取





当极点数为 30 和 50 时，对 canny 结果边缘提取

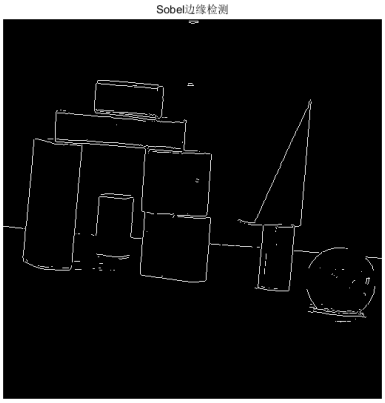


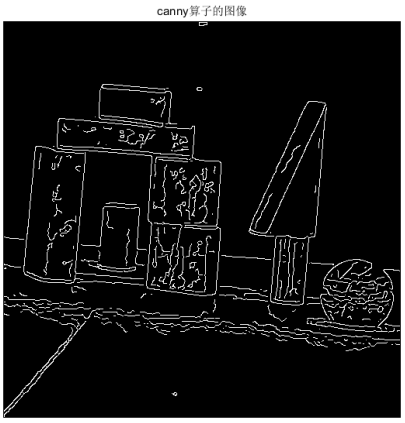


四、结果分析

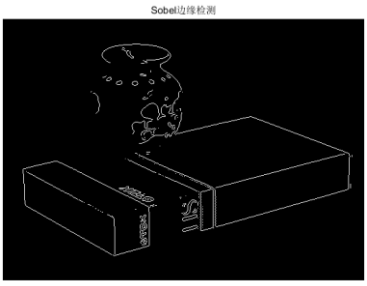
从上面六副作业图得到的结果来看首先是对三种边缘检测的效果来说

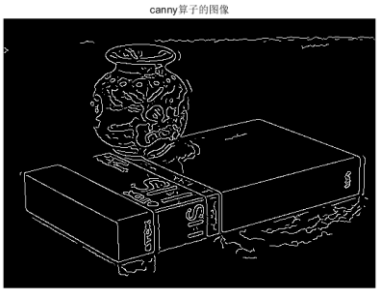
1. sabel 和 prewitt 边缘检测得到结果比较简单，这也可能因为他们的算子针对水平和垂直的两个方向较为敏感，这是因为他们是根据 3*3 的掩模进行卷积得到结果，除了垂直和水平方向，其余方向并不是各向同性的。所以得到的结果较为简单。两种方法的结果基本一致。
2. 而 Canny 边缘检测是比较先进的边缘检测，其算法的先进性，使得局部的边缘，或者说较小的边缘也可以得到显示，我们不妨去对比一下 Test2



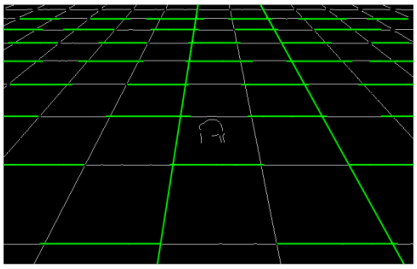


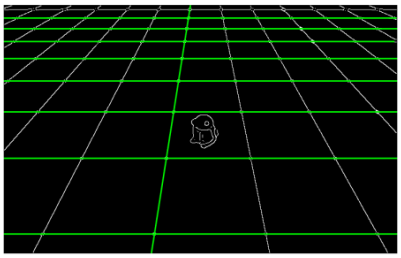
除了更细节的展示外，我们能够发现对于原图左下角 45° 的线条，虽然灰度与周围环境相符但 canny 边缘检测能够检测出来，而 sabel 却没能展示出来/
同样 test3 中，上边部分的暗影也可以展示的很清楚





如果不够清楚的话我们对比 test4 经 huff 边缘提取得出的结果，针对交点数为 30 的情况如下。





对于极点数的讨论我们放到一会将，但是对于同样简单的图像，视觉上两者大体相似，

但是哈夫变换的结果我们能够看到，即使在视觉上看来是边缘的线，但经过哈夫变换后，sabel 边缘检测结果并不如 canny，这也说明了 sabel 边缘检测对于其他方向的局限性。对于 sabel 除了水平与垂直方向外这种各项异性的缺点在 test6 中更能体现。

综上，对比于三种方法，canny 边缘检测优越性得以体现。

哈夫变换的参数对结果的影响。经过理论学习我们知道哈夫变换是在ρ和θ域的焦点个数来决定线的个数。调用P = houghpeaks(H,jidian)函数我们通过调整其中的jidian这个参数，来取在ρ和θ域上焦点最多的点作为参考直线。通过多附图的比较我们可以发现，针对于简单的图像jidian值可以取小一些，但是随着jidian值的取大，边缘个数也随之变大（视觉上来说并非严格增大，因为视觉原因，可能几条直线重合在一起，但像素点来看是两条直线）。这是很正常的情况。

对于函数 [H,T,R]=hough(G,'Theta',value)。其中H,T,R分别表示哈夫矩阵、角度θ以及距离ρ，G表示二值边缘图像，单引号内填写'Theta'表示使用的参数方程可以将xy域变换到上述的ρθ域；value指定θ的值的变化范围，默认为-90°到89.5°。当改变value的值为其他不同的范围时，可以指定检测特定方向的直线。

总体来说最终结果较为满意。

五，实验心得

通过本次作业，我巩固了边缘检测方面的知识，锻炼了我的编程能力，学会了 Matlab 中 edge，Hough，Houghlines，houghpeaks 等函数的使用，知道了如何通过 Hough 变换获取图像中的直线，获益匪浅。

附录

参考文献：

1. 数字图像处理：MATLAB 版：本科教学版/（美）Gonzalez,R.C，woods，R.E，Eddins，S.L.著；阮秋琦译.—2 版.—北京：电子工业出版社，2014.1
2. 数字图像处理：第三版/（美）拉斐尔·冈萨雷斯，（美）理查德·伍兹著；阮秋琦等译.——北京：电子工业出版社，2017.5

代码

Bianyuanjiance.m

```
function bianyuanjiance(img1,jiaodian,fangfa)
img_edge1=edge(img1,'sobel');
img_edge2=edge(img1,'prewitt');
img_edge3=edge(img1,'canny');
figure;
imshow(img1);
title('test1Ô-Í¼');

figure;
imshow(img_edge1);
title('Sobel±ßÔµ¼ì²â');

figure;
imshow(img_edge2);
title('prewitt±ßÔµ¼ì²â');

figure;
imshow(img_edge3);
title('cannyËã×ÓµÄÍ¼Ïñ');
if(fangfa == 1)
    jiance=img_edge1;
elseif(fangfa == 2)
    jiance=img_edge2;
elseif(fangfa == 3)
    jiance=img_edge3;
else
    jiance = img_edge1;
end

[H,T,R] =
hough(jiance);%¼ÆËã¶þÖµÍ¼ÏñµÄ±ê×¼»ô·ò±ä»»£¬HÎª»ô·ò±ä»»¾ØÕó£¬T,RÎª¼ÆËã»ô·ò±ä»»µÄ½Ç¶È°Í°ë¾¶Öµ
P  = houghpeaks(H,jiaodian);%ÌáÈ¡¼«Öµµã
x = T(P(:,2));
y = R(P(:,1));
```

```
lines=houghlines(jiance,T,R,P);%ÌáÈ¡ÏßÌ
figure;
imshow(jiance), hold on;
for k = 1:length(lines)
xy = [lines(k).point1; lines(k).point2];
plot(xy(:,1),xy(:,2),'LineWidth',2,'Color','green');%»-³öÏßÌ
end
end
```