

数字图像处理

直线检测作业

摘要

本次报告主要记录直线检测作业中的各项任务完成情况。本次作业以 Matlab 为平台，结合 Matlab 函数编程实现对所给图像文件的相关处理：1. 首先对测试图像（文件名为：test1~test6）进行边缘检测，常用的边缘提取算法包括 sobel, prewitt 和 canny 算法等等。在实现了三种边缘提取算法以后，我逐个分析了各个滤波器的优缺点，进行对比分析，得到了 canny 滤波器的边缘提取效果最好这一结论。2. 在边缘检测的基础上，用 hough 变换检测图中直线；3. 比较不同边缘检测算法（2 种以上）、不同 hough 变换参数对直线检测的影响。以上任务完成后均得到了预期的结果，并对结果进行了分析，比较了不同边缘检测算法和不同 hough 变换参数对直线检测的影响，对边缘检测算法和 hough 变换有了更加深入的理解。

姓 名：魏佳利

班 级：自动化 63

学 号：2160504076

提交日期：2019 年 5 月 16 日

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 canny 算子方法；

1.1 问题分析：

1.1.1 边缘的定义：

在数字图像中，边缘是指图像局部变化最显著的部分，边缘主要存在于目标与目标，目标与背景之间，是图像局部特性的不连续性，如灰度的突变、纹理结构的图标、颜色的图标等。尽管图像的边缘点产生的原因各不相同，但他们都是图形上灰度不连续或灰度急剧变化的点，图像边缘分为阶跃状、斜坡状和屋顶状。

1.1.2 边缘检测的基本方法：

一般图像边缘检测方法主要有如下四个步骤：

图像滤波

传统边缘检测算法主要是基于图像强度的一阶和二阶导数，但导数的计算对噪声很敏感，因此必须使用滤波器来改善与噪声有关的边缘检测器的性能。需要指出的是，大多数滤波器在降低噪声的同时也造成了边缘强度的损失，因此，在增强边缘和降低噪声之间需要一个折衷的选择。

图像增强

增强边缘的基础是确定图像各点邻域强度的变化值。增强算法可以将邻域(或局部)强度值有显著变化的点突显出来。边缘增强一般是通过计算梯度的幅值来完成的。

图像检测

在图像中有许多点的梯度幅值比较大，而这些点在特定的应用领域中并不都是边缘，所以应该用某种方法来确定哪些点是边缘点。最简单的边缘检测判断依据是梯度幅值。

图像定位

如果某一应用场合要求确定边缘位置，则边缘的位置可在子像素分辨率上来估计，边缘的方位也可以被估计出来。近 20 多年来提出了许多边缘检测算子，在这里我们仅讨论集中常见的边缘检测算子。

1.1.3 常见边缘检测算子分析

sobel 算子

Sobel 算子在边缘检测算子扩大了其模版，在边缘检测的同时尽量削弱了噪声。

其模版大小为 3×3 ，其将方向差分运算与局部加权平均相结合来提取边缘。在求取图像梯度之前，先进行加权平均，然后进行差分，加强了对噪声的一致。其主要用于边缘检测，在技术上它是以离散型的差分算子，用来运算图像亮度函数的梯度的近似值，Sobel 算子是典型的基于一阶导数的边缘检测算子，由于该算子中引入了类似局部平均的运算，因此对噪声具有平滑作用，能很好的消除噪声的影响。Sobel 算子对于像素的位置的影响做了加权，与 Prewitt 算子、Roberts 算子相比因此效果更好。

Sobel 算子包含两组 3×3 的矩阵，分别为横向及纵向模板，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。

prewitt 算子

Prewitt 算子是一种一阶微分算子的边缘检测，利用像素点上下、左右邻点的灰度差，在边缘处达到极值检测边缘，去掉部分伪边缘，对噪声具有平滑作用。其原理是在图像空间利用两个方向模板与图像进行邻域卷积来完成的，这两个方向模板一个检测水平边缘，一个检测垂直边缘。

经典 Prewitt 算子认为：凡灰度新值大于或等于阈值的像素点都是边缘点。即选择适当的阈值 T ，若 $P(i,j) \geq T$ ，则 (i,j) 为边缘点， $P(i,j)$ 为边缘图像。这种判定是欠合理的，会造成边缘点的误判，因为许多噪声点的灰度值也很大，而且对于幅值较小的边缘点，其边缘反而丢失了。

Prewitt 算子对噪声有抑制作用，抑制噪声的原理是通过像素平均，但是像素平均相当于对图像的低通滤波，所以 Prewitt 算子对边缘的定位不如 Roberts 算子。

因为平均能减少或消除噪声，Prewitt 梯度算子法就是先求平均，再求差分来求梯度。

canny 算子

该算子功能比前面几种都要好，但是它实现起来较为麻烦，Canny 算子是一个具有滤波，增强，检测的多阶段的优化算子。

利用 Canny 算子检测边缘的土体算法如下：

- (1) 用式所示的高斯函数 $h(r)$ 对图像进行平滑滤波，去除图像中的噪声。
- (2) 在每一点计算出局部梯度和边缘方向，可以利用 Sobel 算子、Prewitt 算子等来计算。边缘点定义为梯度方向上其强度局部最大的点。
- (3) 对梯度进行“非极大值抑制”。在第二步中确定的边缘点会导致梯度幅度图像中出现脊。然后用算法追踪所有脊的顶部，并将所有不在脊的顶部的像素设为零，以便在输出中给出一条细线。
- (4) 双阈值化和边缘连接。脊像素使用两个阈值 $T1$ 和 $T2$ 做阈值处理，其中 $T1 < T2$ 。值大于 $T2$ 的脊像素称为强边缘像素， $T1$ 和 $T2$ 之间的脊像素称为弱边缘像素。由于边缘阵列孔是用高阈值得到的，因此它含有较少的假边缘，但同时也损失了一些有用的边缘信息。而边缘阵列 $T1$ 的阈值较低，保留了较多信息。因此，可以以边缘阵列 $T1$ 为基础，用边缘阵列 $T2$ 进行补充连接，最后得到边缘图像。

Canny 算子也存在不足之处：

- (1) 为了得到较好的边缘检测结果，它通常需要使用较大的滤波尺度，这样容易丢失一些细节
- (2) Canny 算子的双阈值要人为的选取，不能够自适应

1.2 实验过程：

```
p=imread('test6.jpg');  
p1=edge(p,'sobel');  
p2=edge(p,'prewitt');  
p3=edge(p,'canny');
```

通过 MATLAB 中自带的 edge 函数分别对图像进行 sobel、prewitt 和 canny 边缘检测，得到边缘检测后的图像，比较不同的边缘检测算法，得到各自的优缺点。

1.3 实验结果：

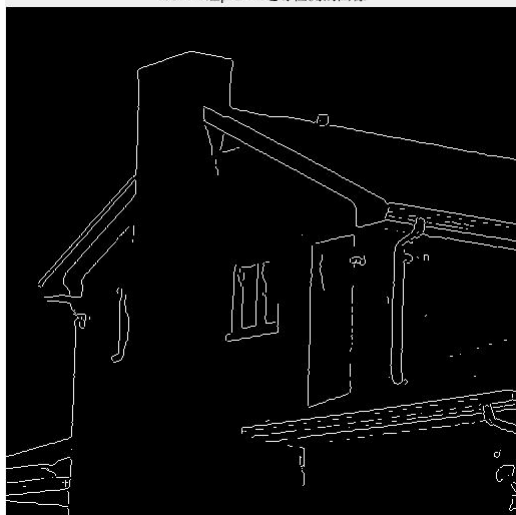
test1.tif的原始图像



test1.tif经Sobel边缘检测的图像



test1.tif经prewitt边缘检测的图像



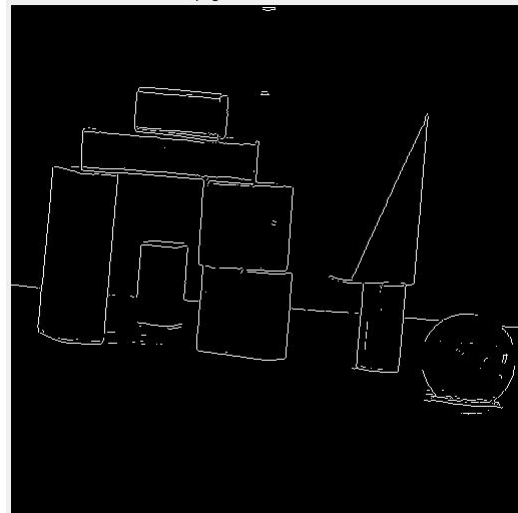
test1.tif经canny边缘检测的图像



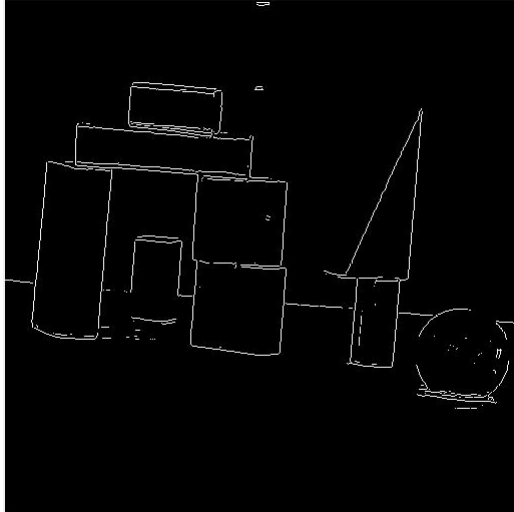
test2.png的原始图像



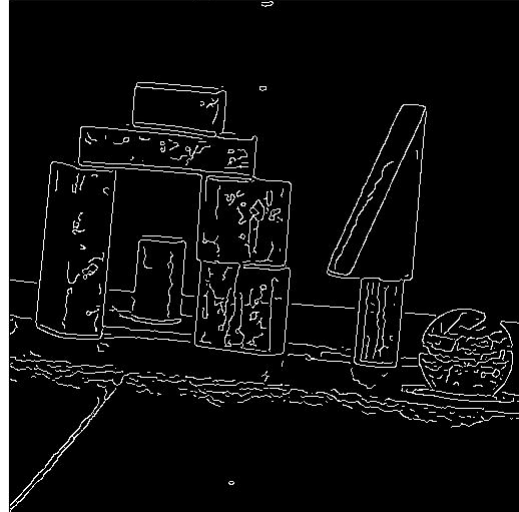
test2.png经Sobel边缘检测的图像



test2.png经prewitt边缘检测的图像



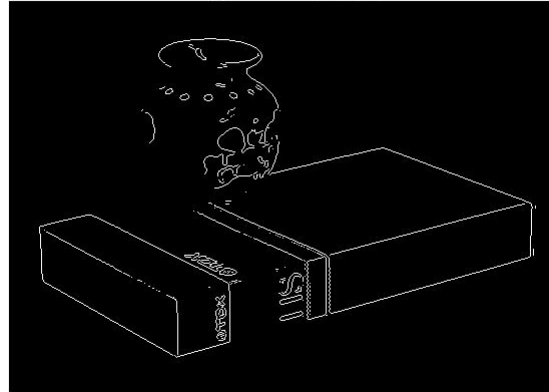
test2.png经canny边缘检测的图像



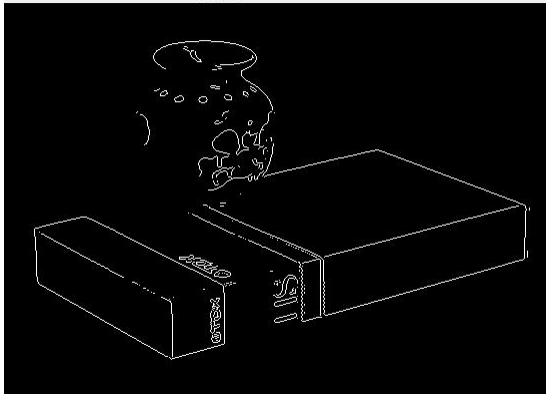
test3.jpg的原始图像



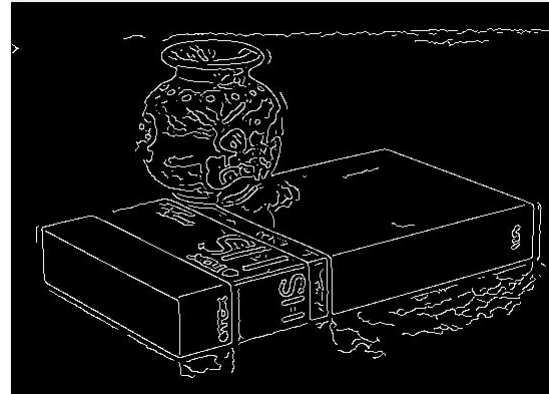
test3.jpg经Sobel边缘检测的图像



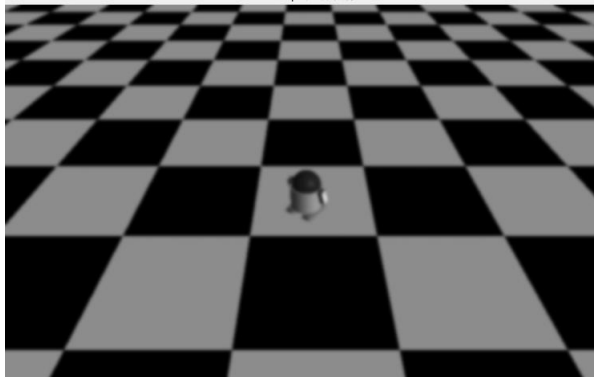
test3.jpg经prewitt边缘检测的图像



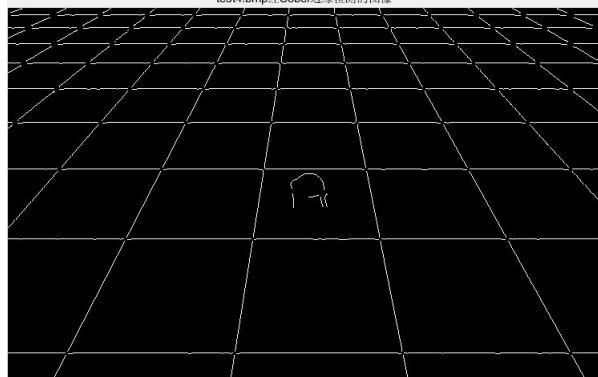
test3.jpg经canny边缘检测的图像



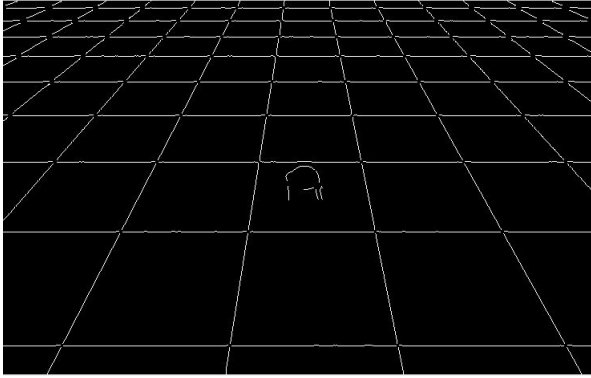
test4.bmp的原始图像



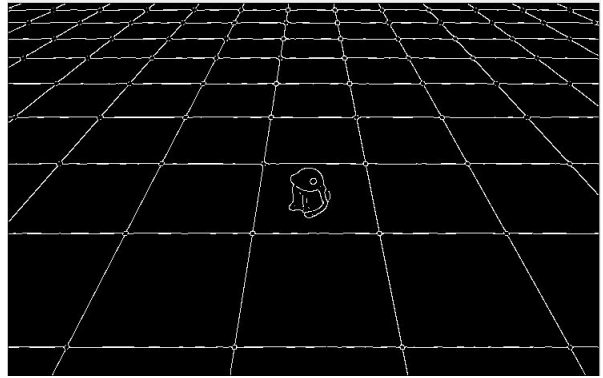
test4.bmp经Sobel边缘检测的图像



test4.bmp经prewitt边缘检测的图像



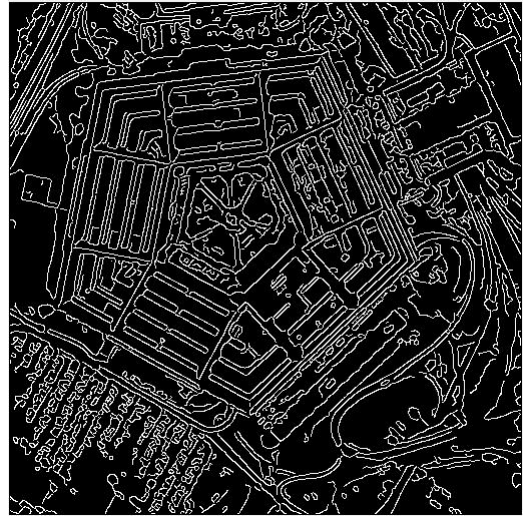
test4.bmp经canny边缘检测的图像



test5.png的原始图像



test5.png经canny边缘检测的图像



test5.png经Sobel边缘检测的图像



test5.png经prewitt边缘检测的图像



test6.jpg的原始图像



test6.jpg经Sobel边缘检测的图像



test6.jpg经prewitt边缘检测的图像



test6.jpg经canny边缘检测的图像



1.4 结果分析:

通过对比每组图像得到的 sobel 和 prewitt 边缘检测结果,可以得到结论:sobel 得到的边缘检测结果含有的噪声比 prewitt 边缘检测结果少,所得到的结果更加平滑。分析可得正是因为他们之间计算上的小差别导致了这一结果,sobel 模板更够较好地抑制平滑噪声。Canny 边缘检测算法得到的边缘更加平滑,细节也更加丰富,整体的效果是最好的。

2、在边缘检测的基础上，用 hough 变换检测图中直线；

1.1 问题分析：

Hough 变换是一种使用表决原理的参数估计技术。其原理是利用图像空间和 Hough 参数空间的点~线对偶性，把图像空间中的检测问题转换到参数空间。

Hough 直线检测的基本原理在于利用点与线的对偶性，在我们的直线检测任务中，即图像空间中的直线与参数空间中的点是一一对应的，参数空间中的直线与图像空间中的点也是一一对应的。这意味着我们可以得出两个非常有用的结论：

- (1) 图像空间中的每条直线在参数空间中都对应着单独一个点来表示；
- (2) 图像空间中的直线上任何一部分线段在参数空间对应的是同一个点。

因此 Hough 直线检测算法就是把在图像空间中的直线检测问题转换到参数空间中对点的检测问题，通过在参数空间里寻找峰值来完成直线检测任务。

如前所述，霍夫直线检测就是把图像空间中的直线变换到参数空间中的点，通过统计特性来解决检测问题。具体来说，如果一幅图像中的像素构成一条直线，那么这些像素坐标值 (x, y) 在参数空间对应的曲线一定相交于一个点，所以我们只需要将图像中的所有像素点（坐标值）变换成参数空间的曲线，并在参数空间检测曲线交点就可以确定直线了。

在理论上，一个点对应无数条直线或者说任意方向的直线，但在实际应用中，我们必须限定直线的数量（即有限数量的方向）才能够进行计算。

因此，我们将直线的方向 θ 离散化为有限个等间距的离散值，参数 ρ 也就对应离散化为有限个值，于是参数空间不再是连续的，而是被离散量化为一个个等大小网格单元。将图像空间（直角坐标系）中每个像素点坐标值变换到参数空间（极坐标系）后，所得值会落在某个网格内，使该网格单元的累加计数器加 1。当图像空间中所有的像素都经过霍夫变换后，对网格单元进行检查，累加计数值最大的网格，其坐标值 (ρ_0, θ_0) 就对应图像空间中所求的直线。

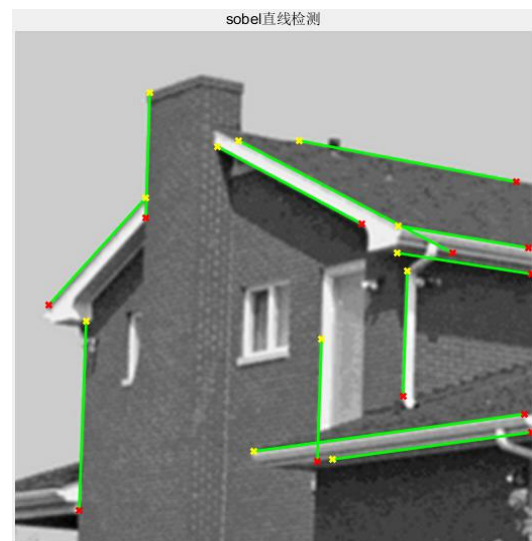
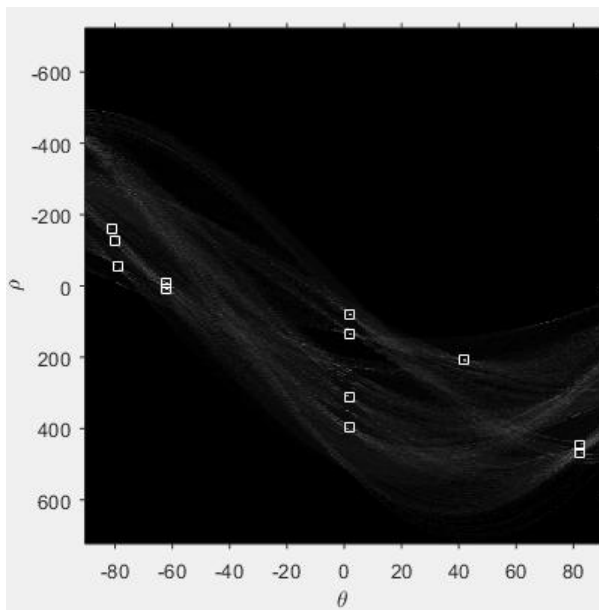
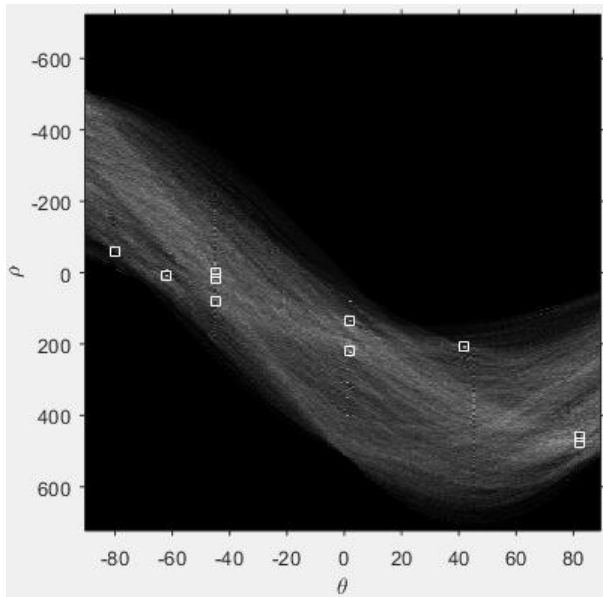
1.2 实验过程：

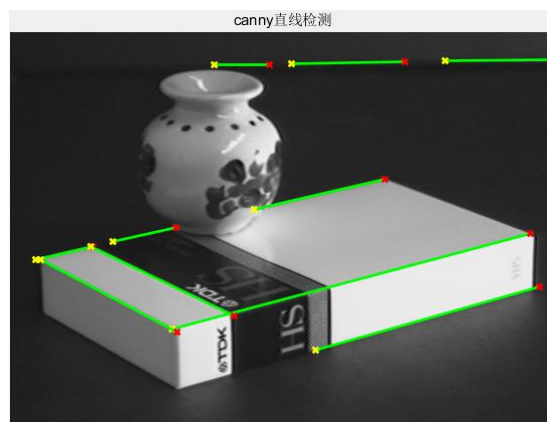
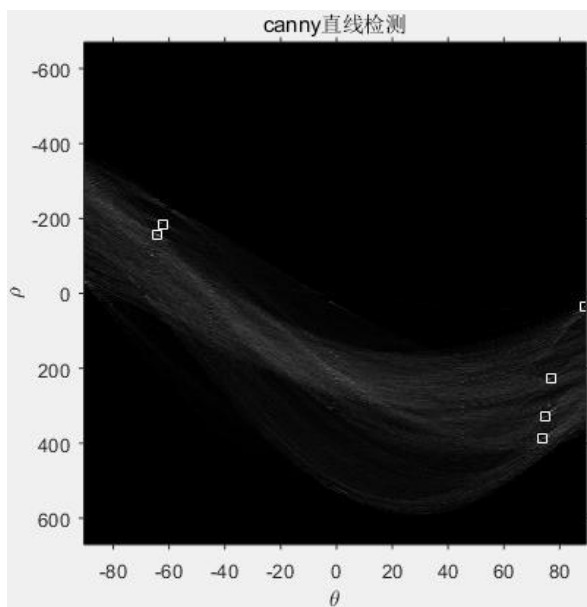
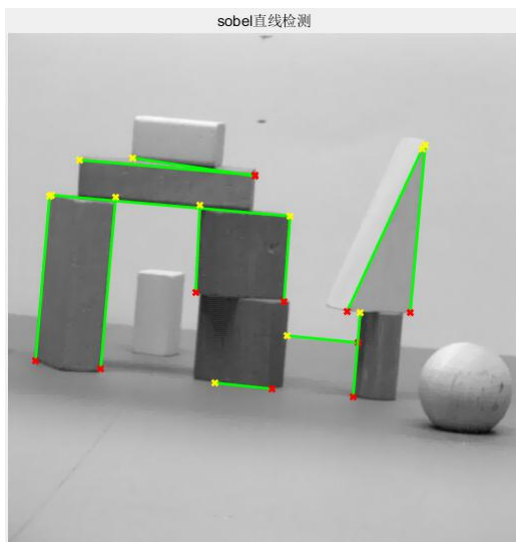
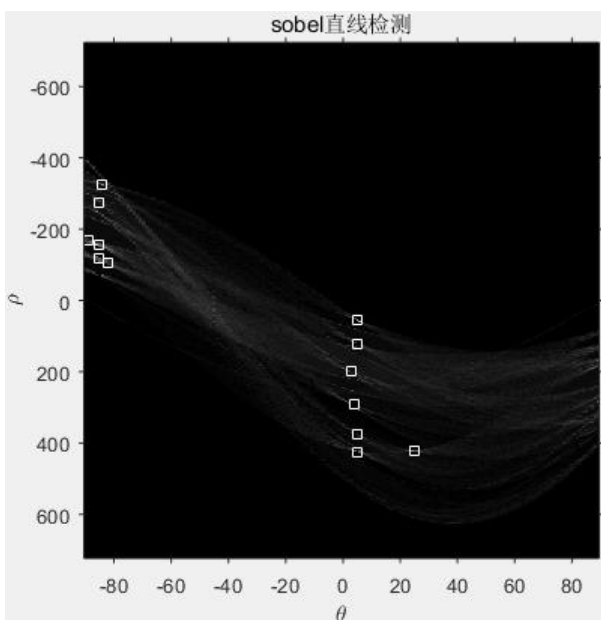
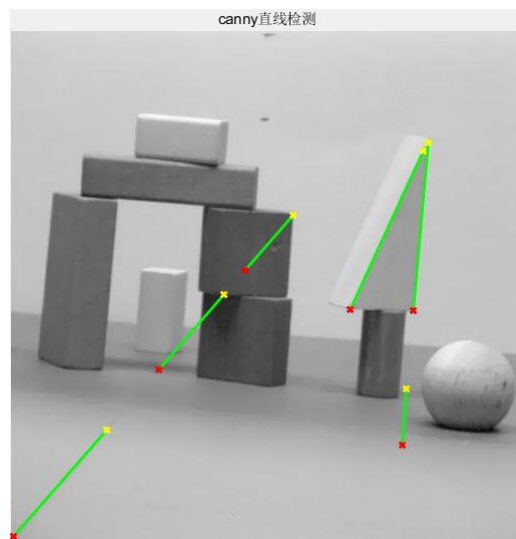
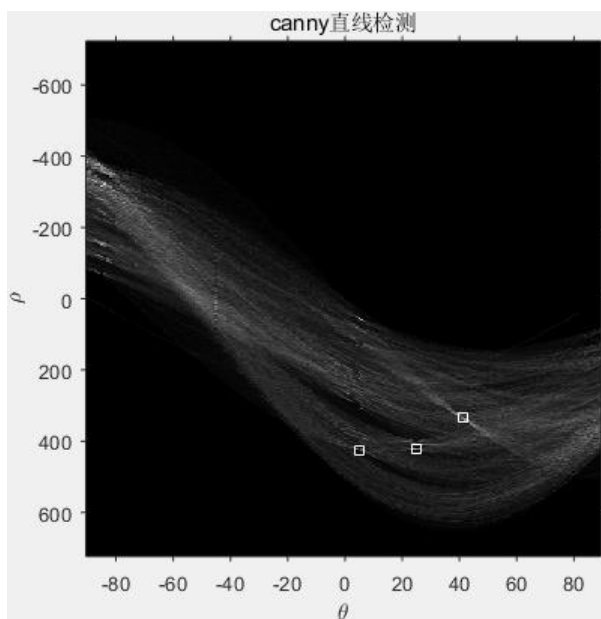
```
[H,T,R] = hough(BW); %计算二值图像的标准霍夫变换，H 为霍夫变换矩阵，T,R 为计算霍夫变换的角度和半径值
```

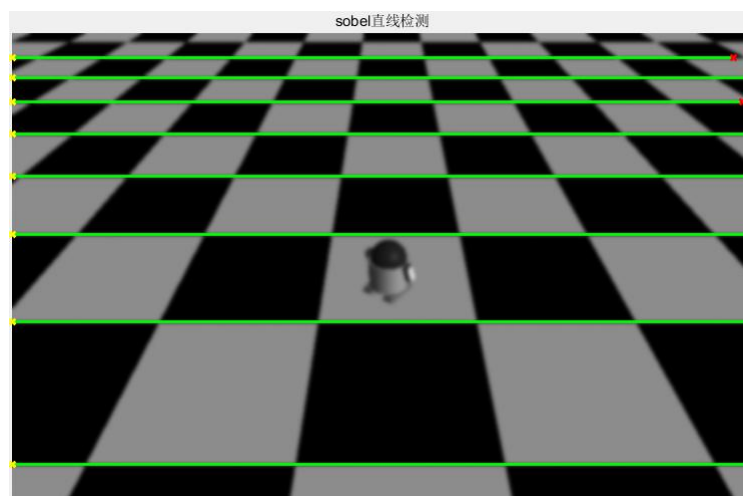
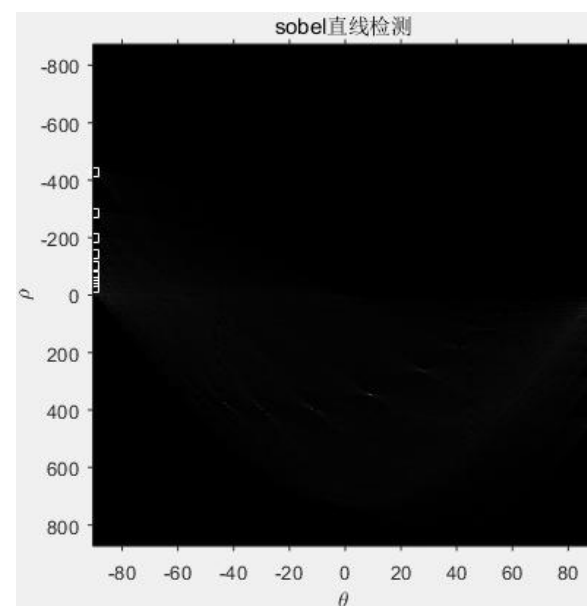
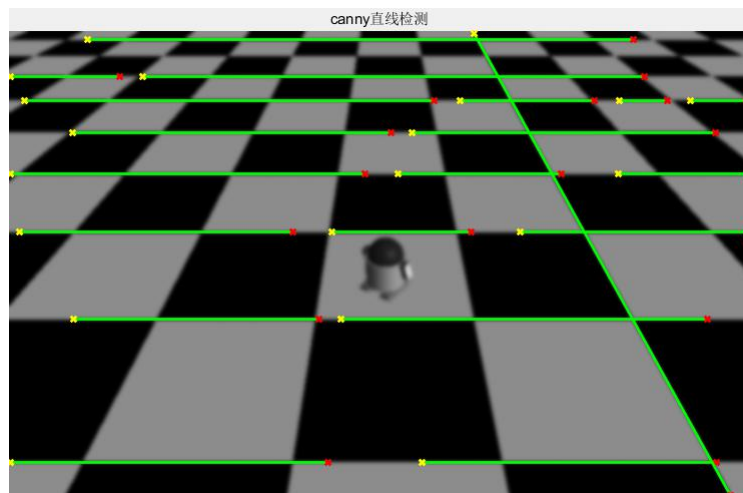
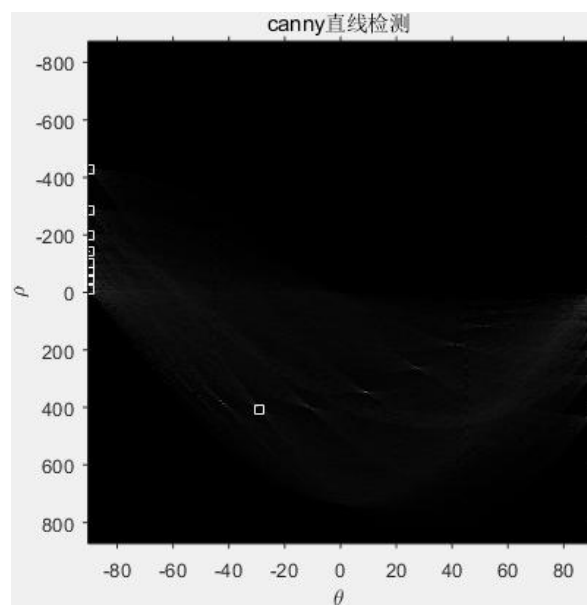
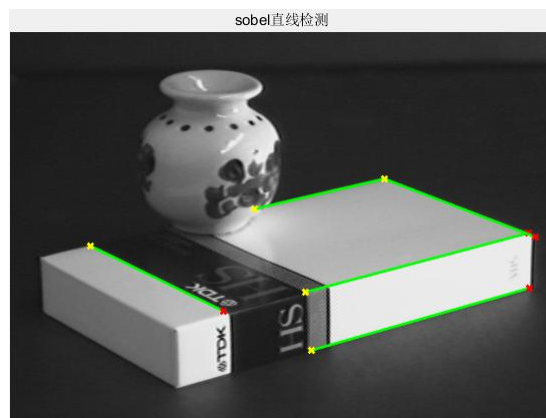
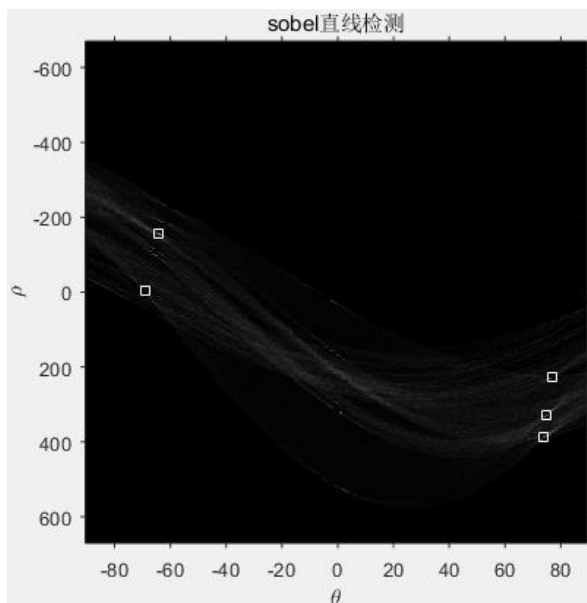
```
P = houghpeaks(H,5); %提取极值点
```

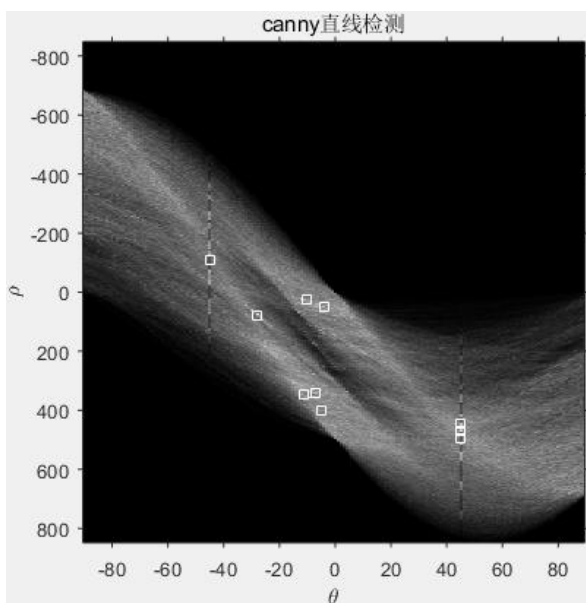
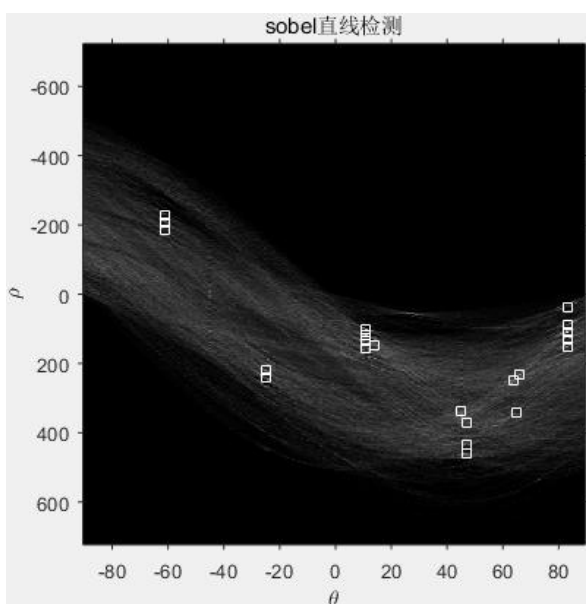
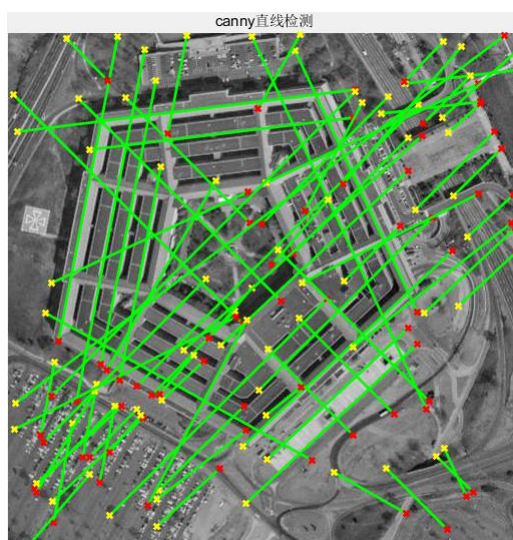
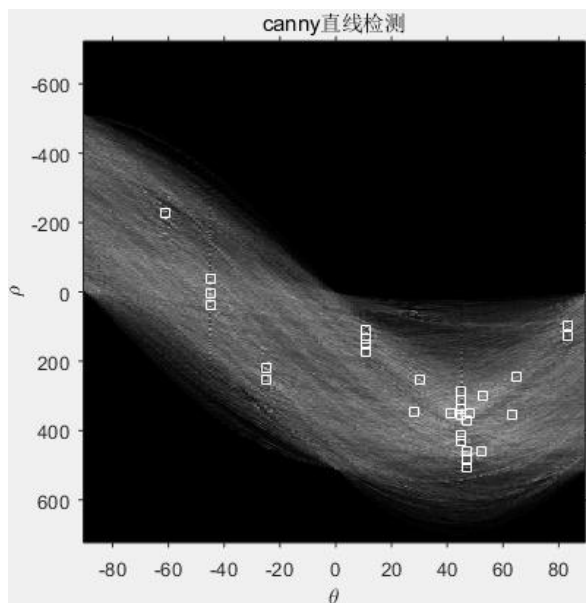
在 sobel 和 canny 边缘检测的基础之上进行霍夫变换，利用 MATLAB 中自带的 hough 函数以及 houghpeaks 函数得到霍夫变换直线检测的结果。通过观察直线检测的结果，得到不同边缘检测算法对直线检测的影响以及提取的极值点数不同对直线检测结果的影响。

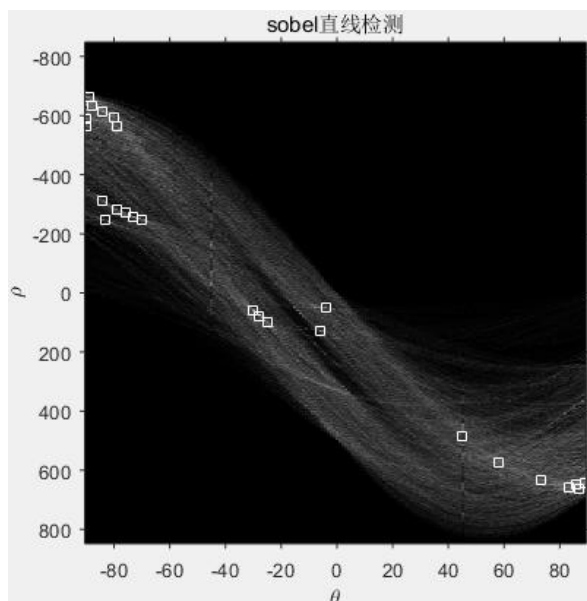
1.3 实验结果:











1.4 结果分析:

用 sobel 边缘检测方法, 进行直线检测时, 可以发现其往往不能检测出所有的直线。在调节极值点个数时可以发现, 往往极值点个数很容易达到上限、饱和。也就是说, 其检测出的直线只是一部分。

用 canny 边缘检测方法, 进行直线检测时, 可以发现其可以检测出大量的直线。只要图像中有一点微小的直线, 它都可以检测出来, 在调节极值点个数时可以发现, 极值点个数不容易达到上限、饱和。也就是说, 它可以检测出来大量的直线, 边缘效果更好。但带来的问题是, 一些视觉上认为不是直线的线也会被检测出来, 带来误差。

3、比较不同边缘检测算法（2 种以上）、不同 hough 变换参数对直线检测的影响;

canny 边缘检测算法得到的边缘较为平滑, 并且得到的细节也很丰富。直线检测结果接近真实但有些直线不易被检出, 比如 test5 中六边形花园的左下角那一条边, 需要更多极大值点才能检测到; 直线检测的结果也容易将视觉上认为非直线的部分检测上去, 比如 test1。

Sobel 边缘检测算法得到的边缘没有 canny 那么平滑, 并且会出现很多断裂。因此会造成一些图像中较为不明显的边缘检测不出来, 进而检测不出来直线。使得增大极值点数容易达到上限和饱和。

在一定范围内, 霍夫变换后选中的极大值点越多, 可以检测出更多的直线。对 test6 的处理中随着 hough 变换参数的增加, 弓箭中的弓形部分甚至可以被逐步还原。但是对于 Sobel 边缘检测算法, 增大极值点数容易达到上限和饱和, 即随着极值点数的增加, 直线检测结果不再发生变化。对于 canny 边缘检测算法,

不断增加极值点数容易将视觉上认为非直线的部分检测上去，带来一定的误差。