

数字图像与视频处理

第七次作业

姓 名：孔维乐

班 级：自动化 65

学 号：2160504122

提交日期：2019 年 5 月

摘要：直线检测是数字图像处理的重要内容，在道路识别，建筑物识别，医学图像分析等领域都有十分重要的应用。通过对已经受直线中的间隙和噪声影响较小。在 M a t a b l 获得的图像进行边缘检测，然后用 H o u g h 变换对边缘检测进行直线检测。该方法简单，环境下对该方法进行了仿真实验，并对图像中直线目标进行了检测，结果表明用该方法对图像中直线目标进行检测是快速有效的。

关键词：Hough 变换 边缘检测 直线检测

实验任务

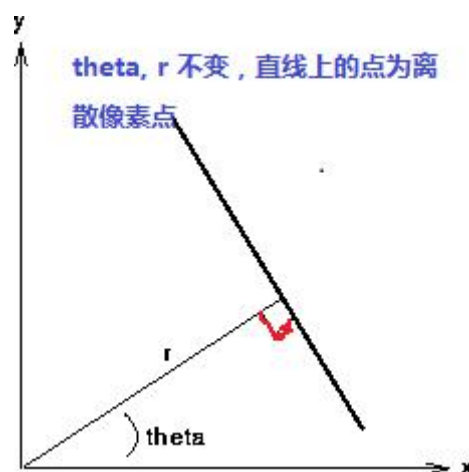
1. 首先对测试图像（文件名为：test1~test6）进行边缘检测，可采用书上介绍的 Sobel 等模板或者 canny 算子方法；
2. 在边缘检测的基础上，用 hough 变换检测图中直线；
3. 比较不同边缘检测算法（2 种以上）、不同 hough 变换参数对直线检测的影响；

实验原理

以直线检测为例，每个像素坐标点经过变换都变成都直线特质有贡献的统一度量，一个简单的例子如下：一条直线在图像中是一系列离散点的集合，通过一个直线的离散极坐标公式，可以表达出直线的离散点几何等式如下：

$$X * \cos(\theta) + y * \sin(\theta) = r$$

其中角度 θ 指 r 与 X 轴之间的夹角， r 为到直线几何垂直距离。任何在直线上点， x ， y 都可以表达，其中 r ， θ 是常量。该公式图形表示如下：

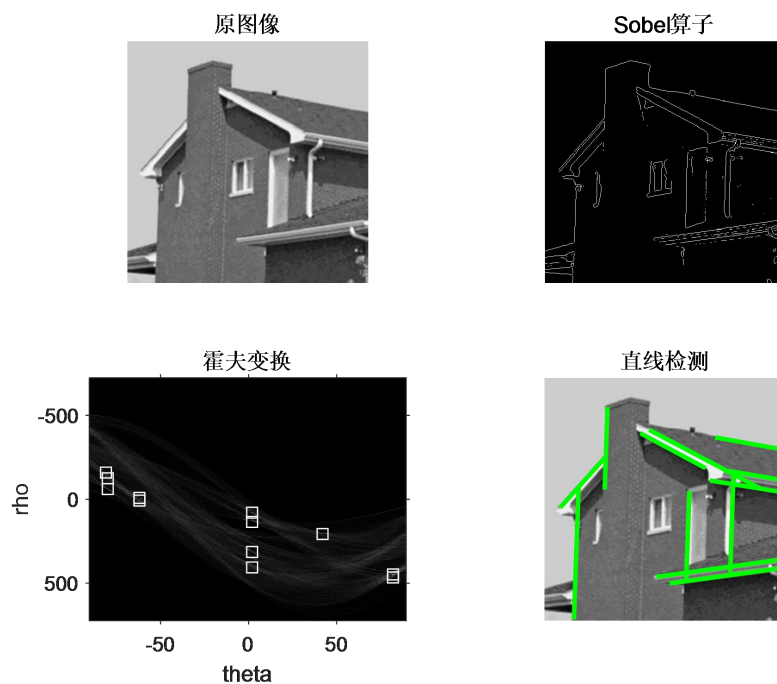


实验处理流程

1. 读取一幅带处理二值图像，最好背景为黑色。
2. 取得源像素数据
3. 根据直线的霍夫变换公式完成霍夫变换，预览霍夫空间结果
4. 寻找最大霍夫值，设置阈值，反变换到图像 RGB 值空间(程序难点之一)
5. 越界处理，显示霍夫变换处理以后的图像

实验结果

Test1 (sobel, peak=12, fillgap=40, minlength=100)



Test1 (canny, peak=15, fillgap=10, minlength=100)

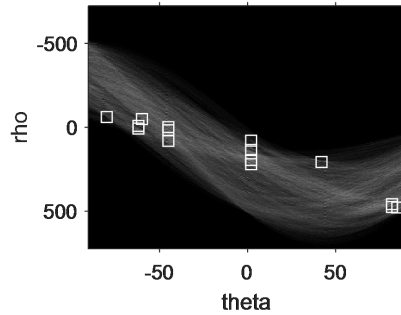
原图像



Canny算子



霍夫变换

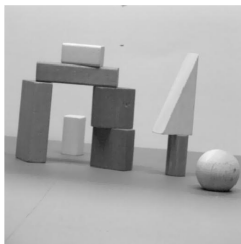


直线检测

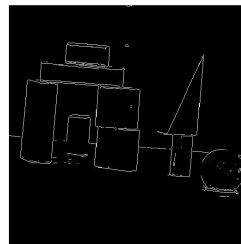


Test2 (sobel, peak=15, fillgap=25, minlength=30)

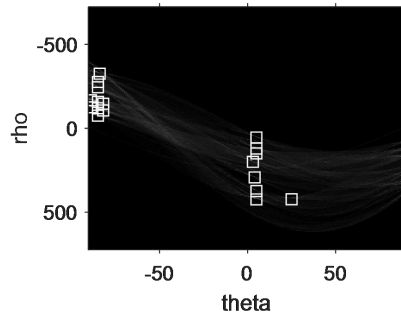
原图像



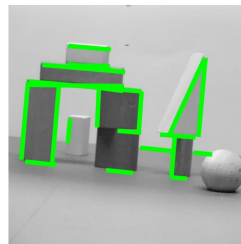
Sobel算子



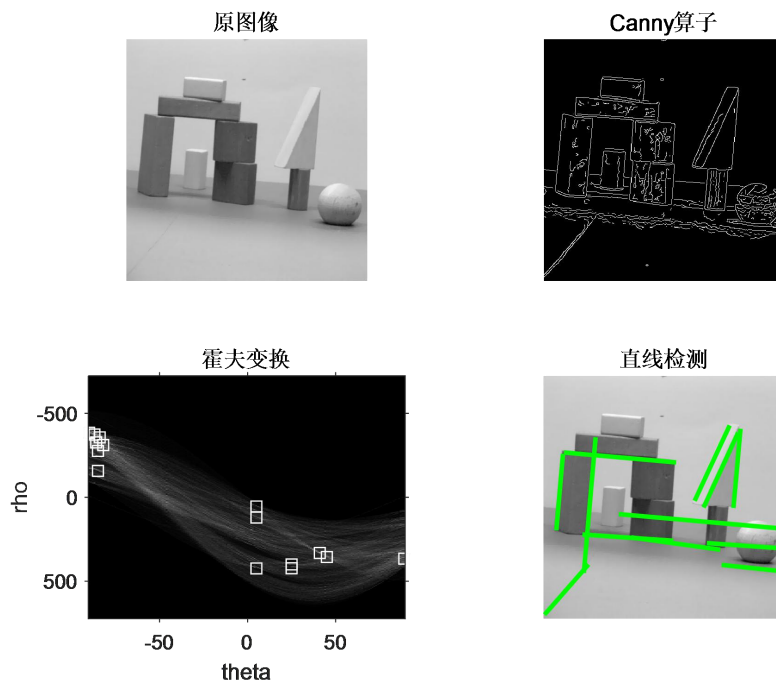
霍夫变换



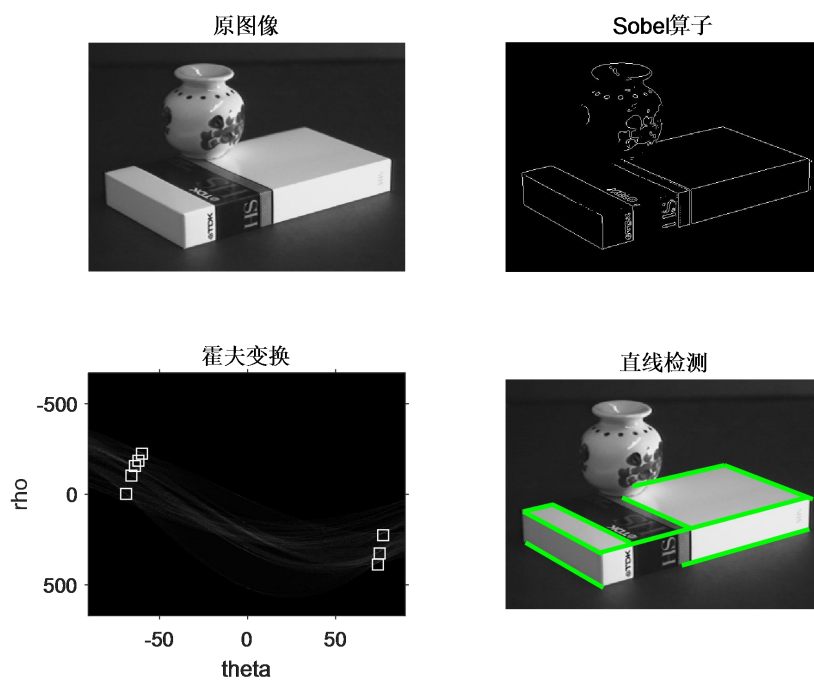
直线检测



Test2 (canny, peak=15, fillgap=25, minlength=120)



Test3 (sobel, peak=8, fillgap=45, minlength=25)

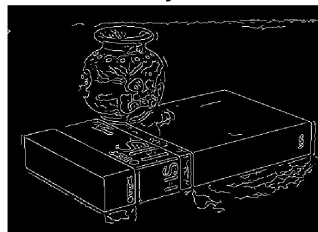


Test3 (canny, peak=10, fillgap=45, minlength=150)

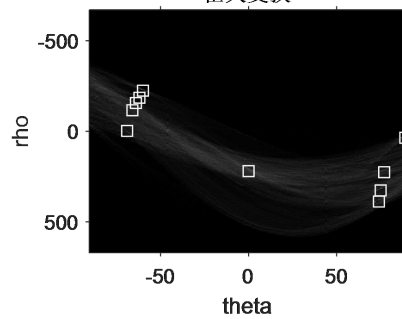
原图像



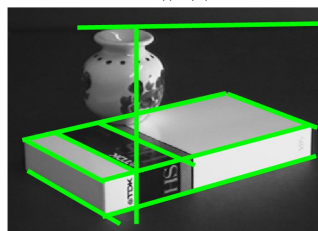
Canny算子



霍夫变换

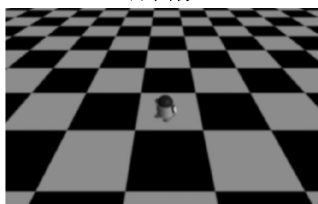


直线检测

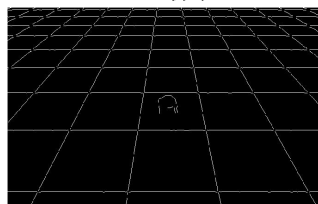


Test4 (sobel, peak=10, fillgap=20, minlength=10)

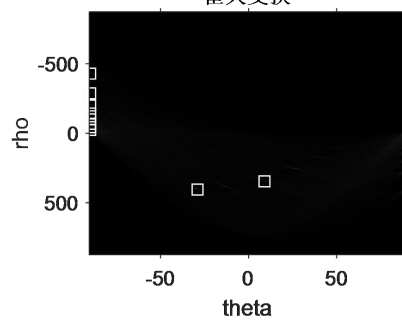
原图像



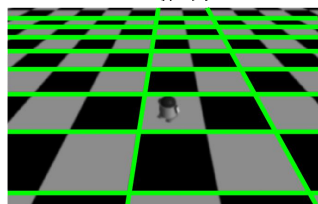
Sobel算子



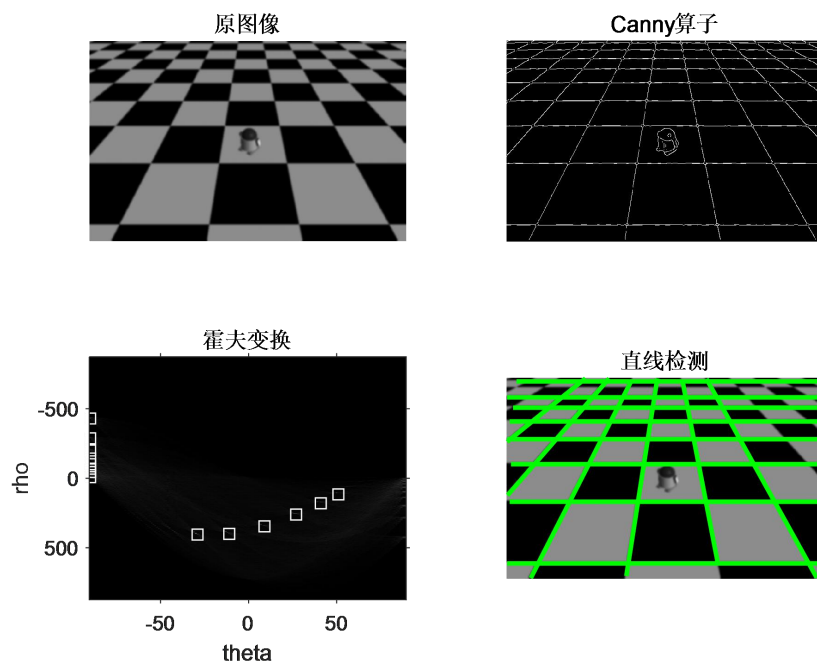
霍夫变换



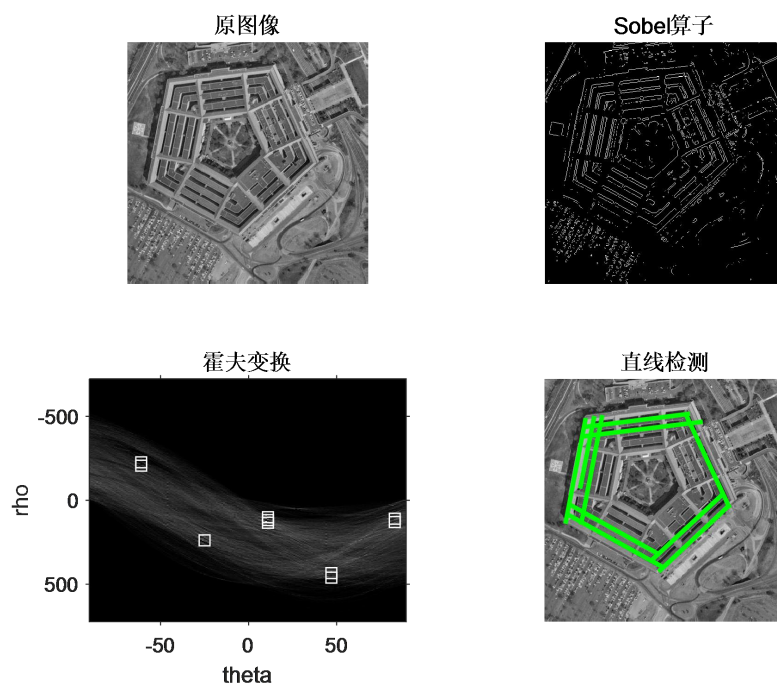
直线检测



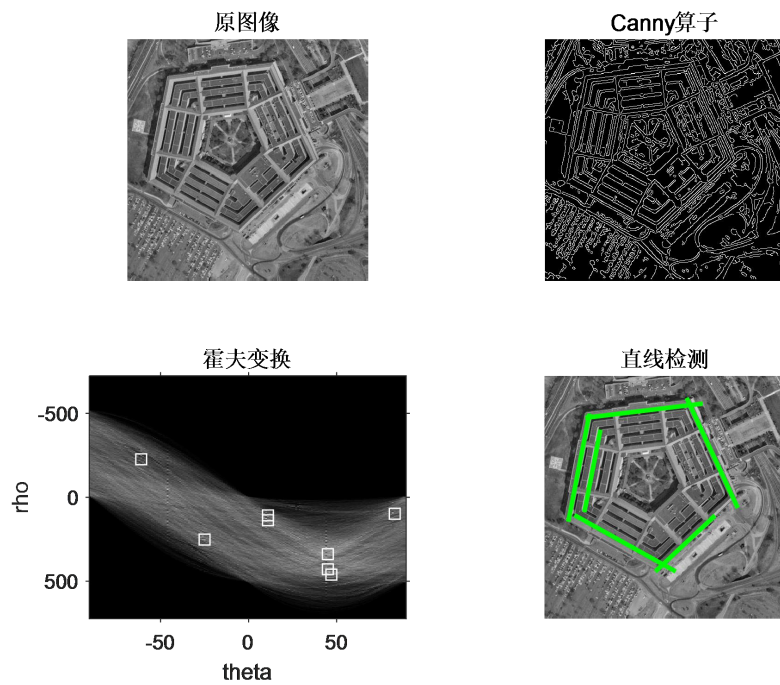
Test4 (canny, peak=15, fillgap=80, minlength=100)



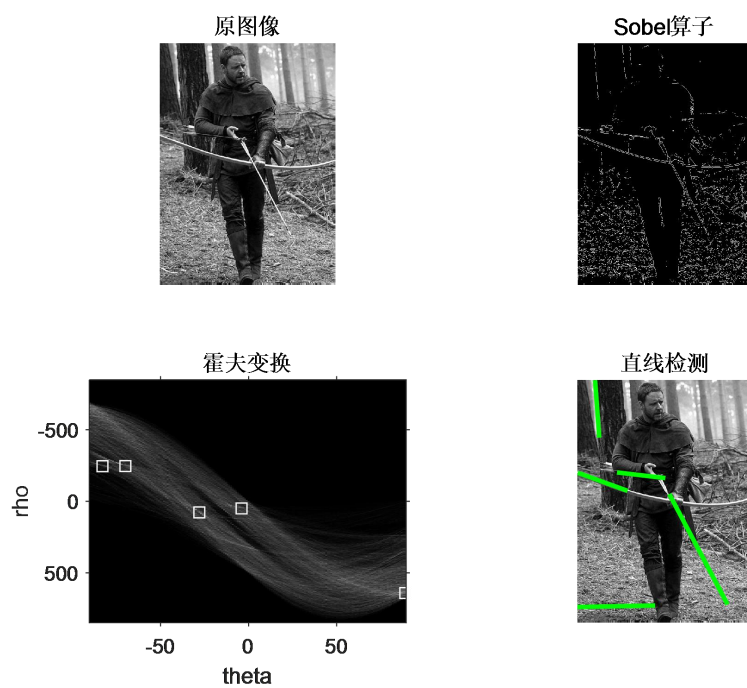
Test5 (sobel, peak=10, fillgap=20, minlength=100)



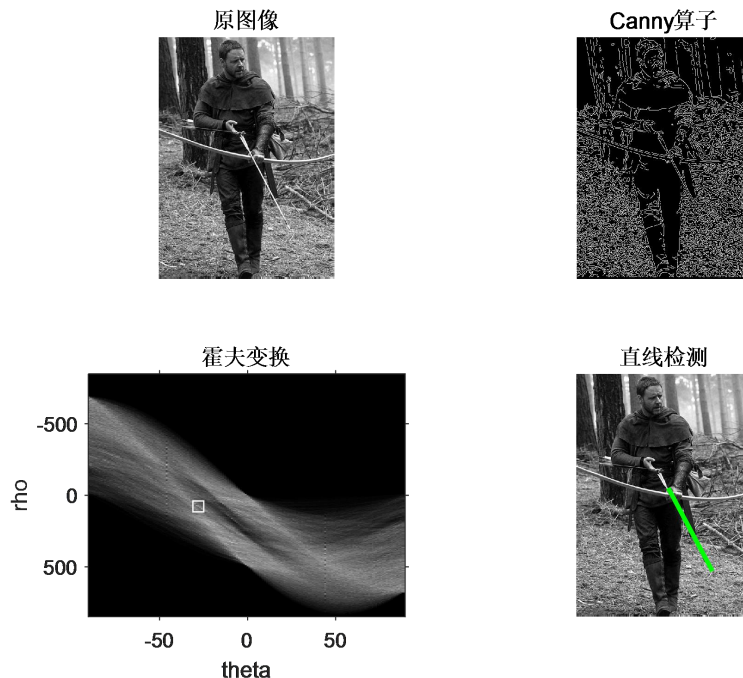
Test5 (canny, peak=8, fillgap=10, minlength=100)



Test6 (sobel, peak=5, fillgap=25, minlength=100)



Test6 (canny, peak=1, fillgap=12, minlength=100)



实验分析

Sobel 算子在空间上比较容易实现，不但产生较好的边缘检测效果，同时，由于其引入了局部平均，使其受噪声的影响也较小。若使用较大的邻域，抗噪性会更好，但也增加了计算量，并且得到的边缘比较粗。在对精度要求不是很高的场合下，Sobel 算子是一种较为常用的边缘检测算法。

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。

由上面的实验结果可以看出，sobel 算子的边缘检测效果较好，canny 算子得到的边缘含有较多的噪声。

对于 Hough 变换中的参数极值 **peak**，填充断点长度 **fillgap**，直线最短长度 **minlength** 进行不同的选择可以得到不同的边缘检测结果。从上面的实验结果可以看到，我们可以根据由算子得到的二值图像，然后对三个参数赋予合适的数值，用以检测出我们所想要检测出来的边缘直线。

源代码

```
clear all
clc
Im=imread('test6.jpg');
S_Im=edge(Im,'sobel');
L_Im=edge(Im,'log');
C_Im=edge(Im,'canny');
subplot(2,2,1);
imshow(Im);

title('原图像');

% subplot(2,2,2);
% imshow(S_Im);

% title('Sobel 算子');

% subplot(2,2,2);
% imshow(L_Im);

% title('Laplace 算子');

subplot(2,2,2);
imshow(C_Im);

title('Canny 算子');

subplot(2,2,3)
[H,T,R] = hough(C_Im);
imshow(H,[],'XData',T,'YData',R,'InitialMagnification','fit');

title('霍夫变换');

xlabel('theta'), ylabel('rho');
axis on, axis normal, hold on;
P = houghpeaks(H,1,'threshold',ceil(0.3*max(H(:)))));
x = T(P(:,2)); y = R(P(:,1));
plot(x,y,'s','color','white');
% Find lines and plot them
lines =
houghlines(C_Im,T,R,P,'FillGap',12,'MinLength',100);
subplot(2,2,4)
imshow(Im), hold on

title('直线检测');

max_len = 0;
```

```

for k = 1:length(lines)
    xy = [lines(k).point1; lines(k).point2];

plot(xy(:,1),xy(:,2),'LineWidth',2,'Color','green');

    % Plot beginnings and ends of lines
    %
plot(xy(1,1),xy(1,2),'x','LineWidth',2,'Color','yellow');
    %
plot(xy(2,1),xy(2,2),'x','LineWidth',2,'Color','red')
;

    % Determine the endpoints of the longest line segment
    len = norm(lines(k).point1 - lines(k).point2);
    if ( len > max_len)
        max_len = len;
        xy_long = xy;
    end
end

```