复习大纲

[1. 人类视觉VS计算机视觉 2](#_Toc504343104)

[2. Intensity normalisation Vs Histogram equalisation 3](#_Toc504343105)

[2.1 Intensity normalisation （直方图亮度正规化） 3](#_Toc504343106)

[2.2 Histogram equalisation （直方图均衡化） 3](#_Toc504343107)

[3. Edge detection （边缘检测） 6](#_Toc504343108)

[3.1 一阶微分边缘检测 First-order edge detection 6](#_Toc504343109)

[3.2 二阶微分边缘检测 Second-order edge detection 6](#_Toc504343110)

[3.3 一阶基本算子推导 7](#_Toc504343111)

[3.3 Prewitt算子，Sobel 算子推导及其扩展 9](#_Toc504343112)

[3.4 Canny算子性质 11](#_Toc504343113)

[3.4 Laplacian 算子推导 11](#_Toc504343114)

[3.5 Laplacian算子改良方法（Marr-Hildreth算子） 13](#_Toc504343115)

[4. Finding Shapes (Hough Transform霍夫变换) 14](#_Toc504343116)

[4.1 直线霍夫变换 14](#_Toc504343117)

[4.2 Foot-of-normal （极坐标直线霍夫检测） 16](#_Toc504343118)

[4.3 圆的霍夫变换 17](#_Toc504343119)

[5. 离散傅里叶变换(Discrete Fourier Transform) 17](#_Toc504343120)

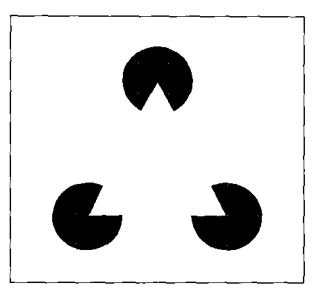
# 人类视觉VS计算机视觉

人类视觉善于区分相对(relative)距离，而不擅长区分绝对（absolute）距离，计算机视觉恰恰相反。

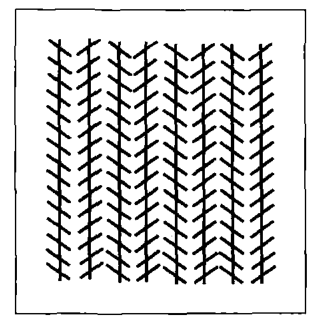
人类视觉描述的也并不是真正的世界. 比如:人眼蓝视锥细胞数量少，大脑通过了某种方式弥补了蓝色感元的不足，使我们感知到了蓝色。而计算机视觉依赖采集的硬件条件。

人类视觉比计算机视觉复杂的多，比如红外制导导弹(Infrared guided-missile)上的计算机视觉系统 很难分辨100M远处的鸟和10km远处的一架飞机（像素比例很小），而人类视觉可以轻松的做到。

人类视觉受立体几何(solid geometry)的先验知识影响，比如下面这个图。可以是一个白色三角形放在三个黑色圆上，或者是三个吃豆人。计算机视觉没有这个先验知识。



人类视觉可以被欺骗，计算机视觉不会被欺骗，比如下面这个图，实际上这些线是直的。



经济成本，计算机视觉只需要摄像机，摄像机界面，计算机就可以具备基本的视觉系统，硬件价格十分低廉（和人类比起来）。

# Intensity normalisation Vs Histogram equalisation

（直方图亮度正规化） Vs (直方图均衡化）

预备知识：

一些图像的直方图统计中表示，没有用到所有的灰度，有些灰度是空的。我们可以通过拉伸图像直方图来利用全部灰度级，效果如下图所示



这样做的好处：

(1) 让图像更加清晰（Better contrast and appears better for human eyes）

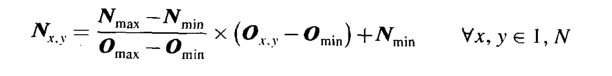
(2) 让图像层次分明

(3) 显示图像中是否有许多噪音

(4) 是很多进一步处理的前置工作

## 2.1 Intensity normalisation （直方图亮度正规化）

将原直方图进行扩展和位移，使直方图涵盖所有256个亮度级。



O – Original （原图像）

N- New (新图像)

公式理解：

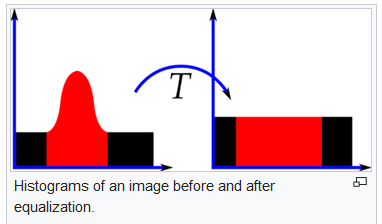
本质上是一个区间map投射函数

将原来Omin – Omax 区间的数字投射到 Nmin -Nmax中，

显然 Nmin = 0, Nmax = 255

## 2.2 Histogram equalisation （直方图均衡化）

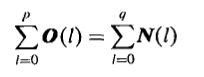
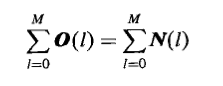
1. 这是一个非线性(Nonlinear)处理过程，不可逆转
2. 目的是通过一种适合人类视觉分析的方法来增强图像的亮度(Highlight image brightness in a way particularly suited to human visual analysis)
3. 直方图均衡化能让图像具有更平坦(flatter)的直方图，所有亮度等概率(equiprobable)的出现。 效果如图下所示



1. 直方图均衡化可以特征增加清晰，如下图所示



公式推导：



O – Original （原图像）

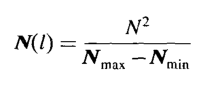
N- New (新图像)

M – 亮度级的数量

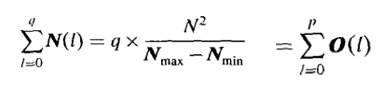
p,q – 亮度级（level） 0-255

首先要保证 亮度与像素数量之和不变

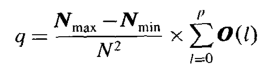
因为输出直方图是平坦的，每个亮度级的数量是固定的，现在可以计算每个亮度级中应该放多少个像素点进去 N(l)，其中N^2是总像素数量

（横着）

因为一共有255个亮度级，所以累加等于乘积

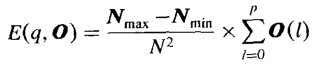


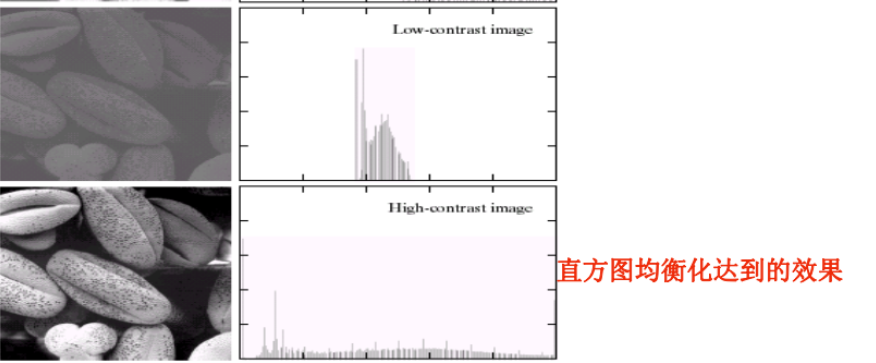
反解得到

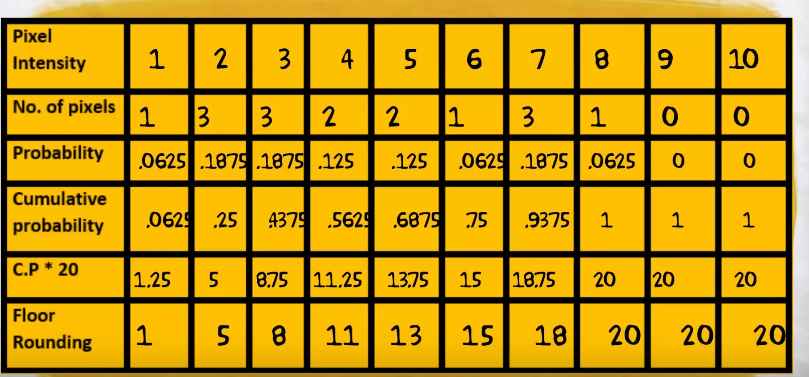


这个是映射函数，所以









对比

Intensity normalisation 和 Histogram equalisation

（直方图亮度正规化）(直方图均衡化）

1. 直方图均衡化的性能的说服力更强（特点在2.2中共有四点）
2. 两个方法都增加了图像的对比度(contrast)
3. 在图像采集过程中的噪音会影响原直方图的形状，会对均衡化产生比较大的影响(can not resist the noise)。
4. 因为噪音问题只能减少而不可以避免，直方图均衡化很少用于通用图像处理中，但是在专业应用中非常有效。
5. 均衡化是非线性处理，图像不可以逆转恢复(return to the original)，而亮度正规化可以逆转恢复图像. （备注:维基百科上说可逆，Mark的书上说不可逆，这里暂时采用Mark的结论）
6. 综上所述，直方图亮度正规化才是常用的正规方法

You cannot avoid noise in electrical systems, however well you design a system to reduce its effect. Accordingly, histogram equalization finds little use in generic image processing systems though it can be potent in specialized applications. For these reasons, intensity normalization is often preferred when a picture’s histogram requires manipulation

# Edge detection （边缘检测）

边缘检测是一种低层次处理(Low-level processing), 不需要任何空间(Spatial)信息

边缘检测强调了图像的对比度, 检测对比度可以增强图像中的边界特性

(Edge detection highlights image contrast, and emphasize the boundaries of features in image)

边界特性就是人类视觉感知目标周边的机制

(It is how human vision can perceive the perimeter of an object)

**边界实际上是亮度级的阶梯变化，而边缘是阶梯变化的位置**

(The boundary of an object is a step change in the intensity levels, and the edge is at the position of the step change)

## 3.1 一阶微分边缘检测 First-order edge detection

一阶微分可以使变化增强，当信号不改变时，一阶微分没有响应

(First-order differentiation since this emphasizes change and it gives no response when applied to signals that do not change)

水平(horizontal)算子可以检测垂直(vertical)方向上的改变，反之

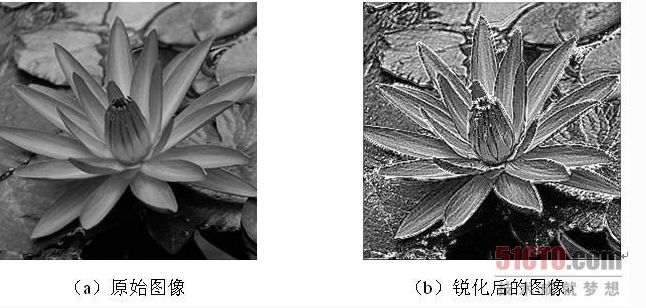
一阶微分算子有 一阶基础算子,Roberts, Prewitt , Sobel 和 Canny

## 3.2 二阶微分边缘检测 Second-order edge detection

一阶微分可以描述图像亮度的增强变化，二阶微分描述了增强的速度。通过找到一阶导数的零点，得到导数最大的点。

二阶微分对噪声也比较敏感，解决的方法是先对图像进行平滑滤波，消除部分噪声，再进行边缘检测

二阶微分算子获得的边界是比较细致的边界。反映的边界信息包括了 许多的细节信息，但是所反映的边界不是太清晰，如下图



二阶微分算子有 Laplacian， Marr-Hildreth

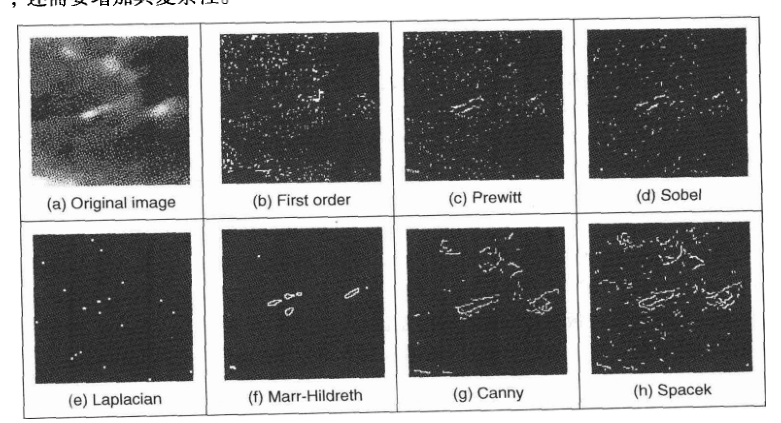
First-order vs Second-order

(1) 定义不同（参考上面）

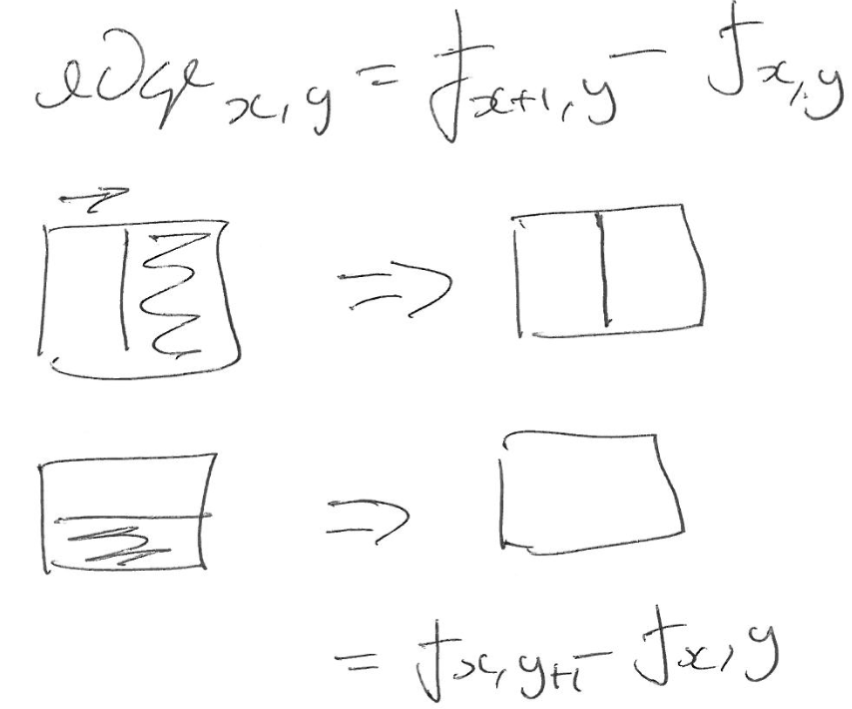
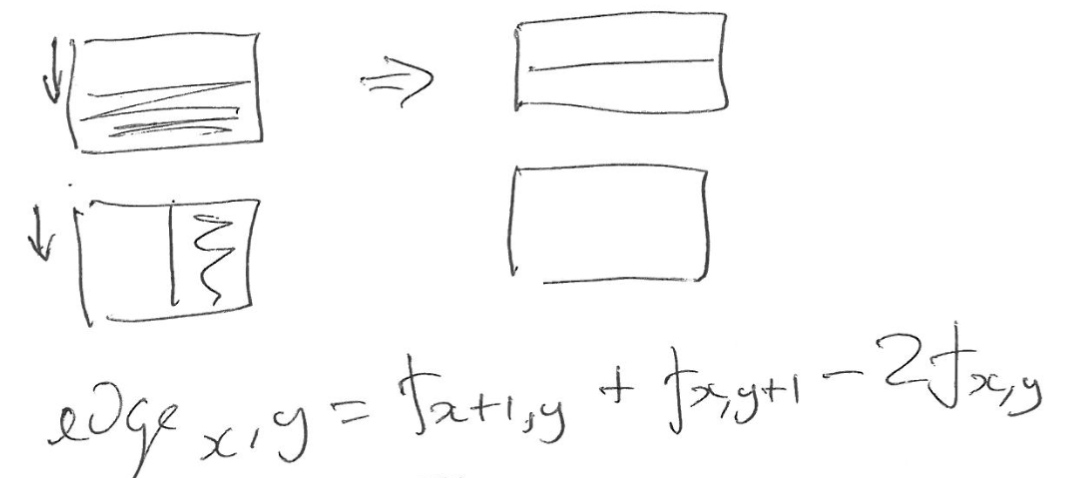
(2) 基础的一阶算子(operator)对噪声的响应非常好，Prewitt和Sobel算子有很多噪音

(3) 二阶算子Laplacian算子几乎没有留下什么信息，Marr-Hildreth处理结果同样很差（二阶算子对噪音敏感）

(4)二阶算子Laplacian只有一个模板，没有Gx和Gy模板，因此得不到边缘的方向



## 3.3 一阶基本算子推导

分析：

泰勒级数对导数定义展开有 O△x是误差



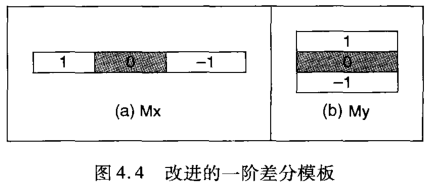


两个相减整理得到



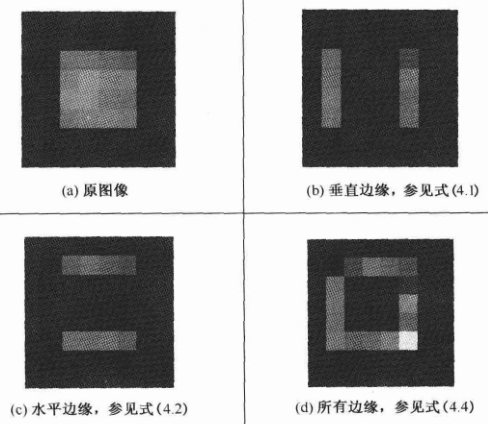
如果△x >> △x^2 时既 △x<<1时，误差会好一些(it is better)

这个公式表明了一阶微分的估算值是由一个像素(pixel)隔开的两个点的差值，误差为 ，因此可以得到改良的一阶差分模板。



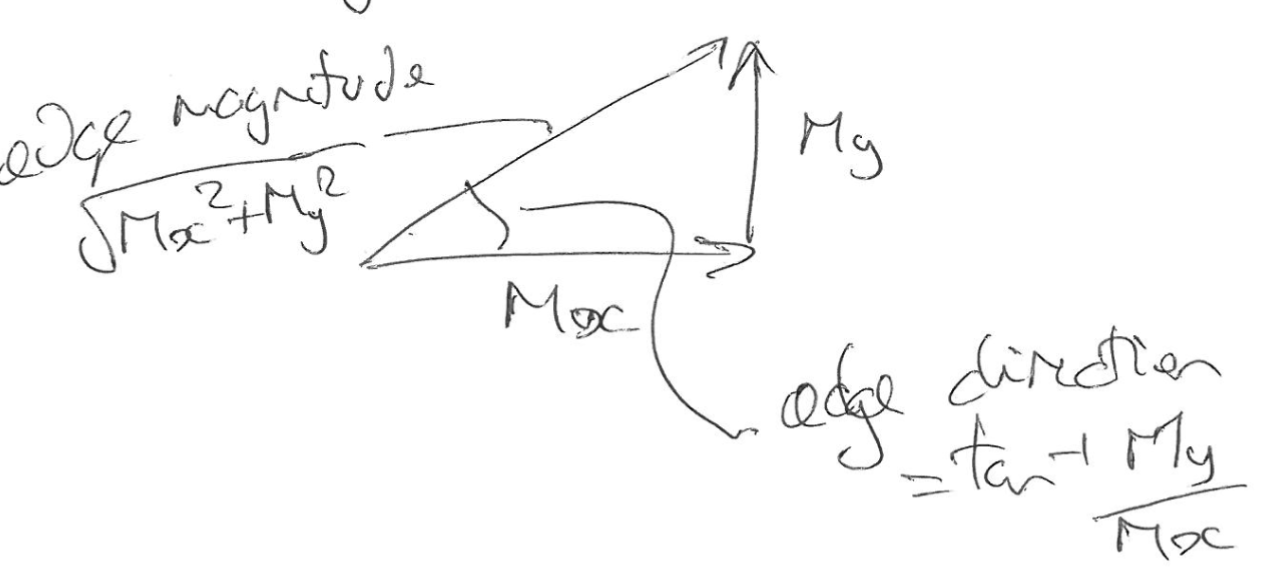
一阶基本算子  可以与图像做卷积来监测边缘点

但是效果比较差，如下图所示，左上角的在正方形点没有被监测到，而右下角的点被检测了两次。因此需要采用均匀阈值方法(Uniform thresholding) 来选取最亮点，改良结果.



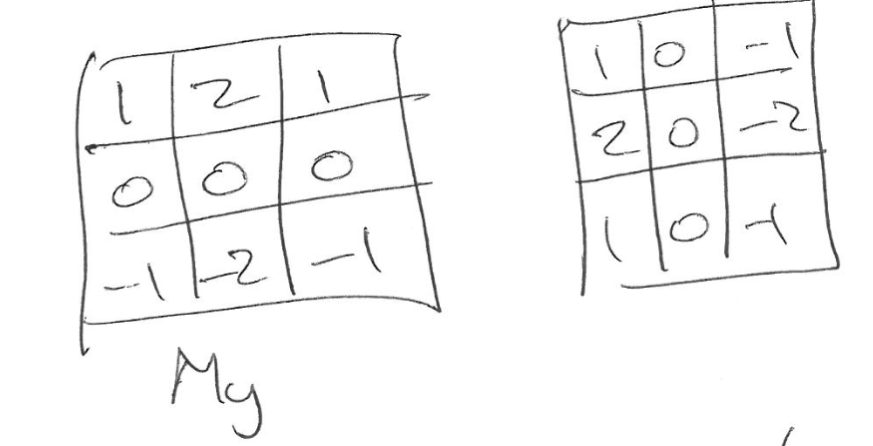
算子的向量分析

边缘强度(edge magnitude)和边缘方向(edge direction)定义如下

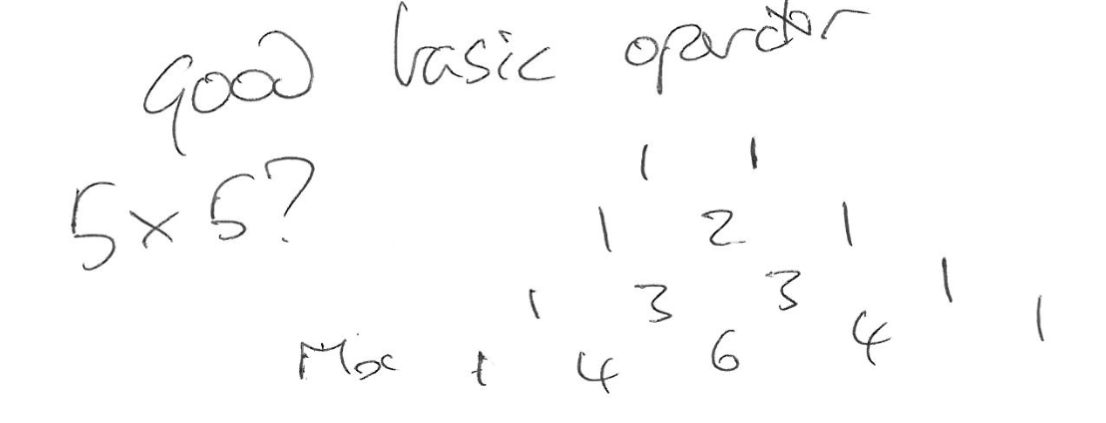


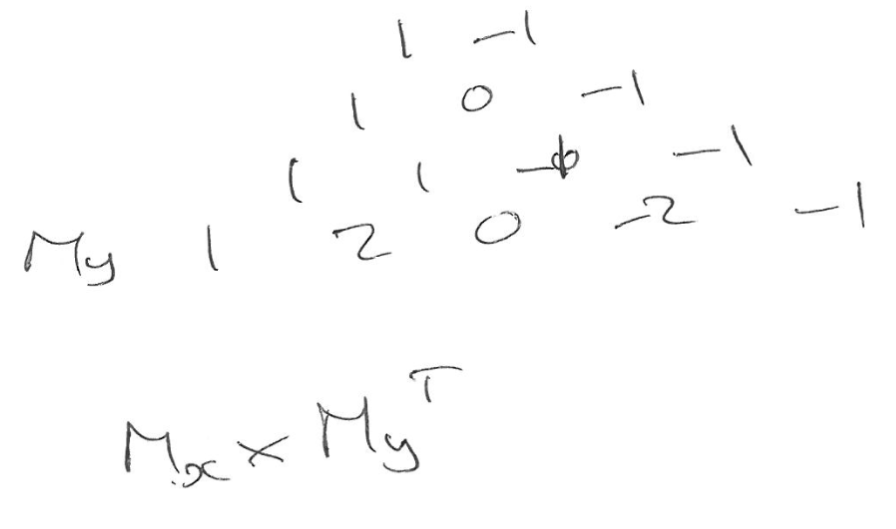
## 3.3 Prewitt算子，Sobel 算子推导及其扩展

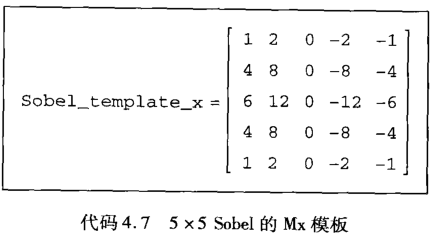
Sobel算子如图下所示，采用了高斯平均 (Employs Gaussian average)



通过如图所示的帕斯卡三角(Pascal's Triangle)可以得到Sobel对应系数

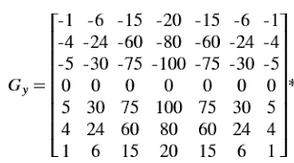






备注：第二列乘了一个2

7X7的Sobel模板

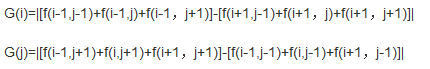


备注：第一行从帕斯卡三角形获得，第二行乘2，第三行是第一行+第二行

乘法系数是由高斯平均 (Employs Gaussian average)决定的（?）

直接推导Sobel算子，需要先推导出Prewitt算子，在3X3模板中

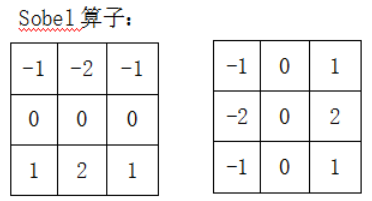
Prewitt算子的定义如下



或者是定义2：将之前提到的改良的一阶差分算子并排扩展成3X3



Sobel算子是在Prewitt算子的基础上改进的，在中心系数上使用一个权值2，相比较Prewitt算子，Sobel模板能够较好的抑制（平滑）噪声

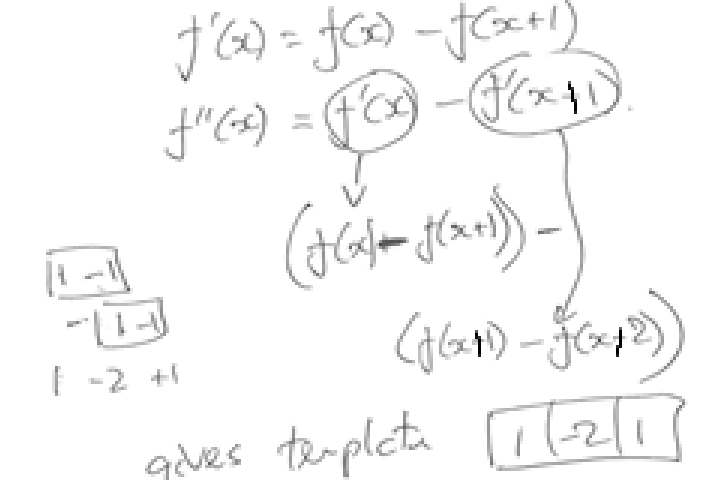


## 3.4 Canny算子性质

Canny算子的特性

1. Optima detection with no spurious response
2. Good Localization with minimal distance between detected and true edge position
3. Single response to eliminate multiple responses to a single edge.

## 3.4 Laplacian 算子推导



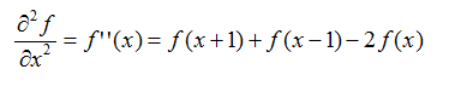
老师的公式推导得到 f(x)-2f(x+1)+f(x+2)

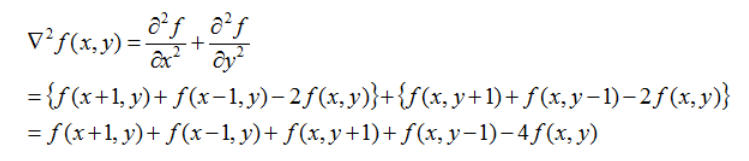
老师是把最左边的方块看成f(x)，通过系数对应可以得到算子

这里进行一个**改良**，将最中间的方块看成f(x) ,从算子逆推可以得到公式

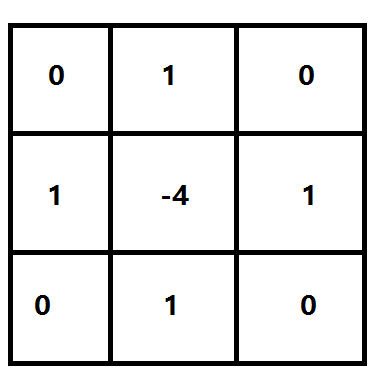
-2f(x) + f(x+1) + f(x-1) （对应系数法） 推荐用这个方法

扩展成二维的形式，做两次偏导





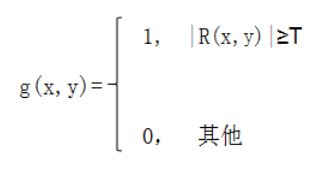
这时你得到的模板为



Laplacian扩展: 将这个算子旋转45°后与原算子相加，就变成八邻域的算 子了



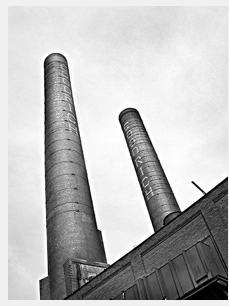
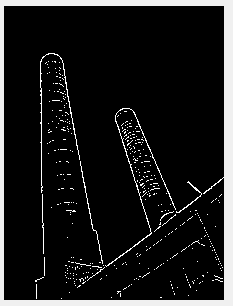
在用Laplacian算子图像进行卷积运算时，当响应的绝对值超过指定阈值时，那么该点就是被检测出来的孤立点，具体输出如下：



备注：用Laplacian算子做边缘检测时应该加入这个条件

做了一个小实验，用的Laplacian算子是 

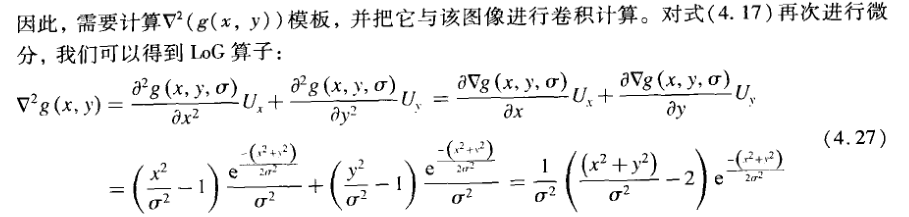
Threshold设置是0.9 看起来加了阈值后边缘更明显了一点

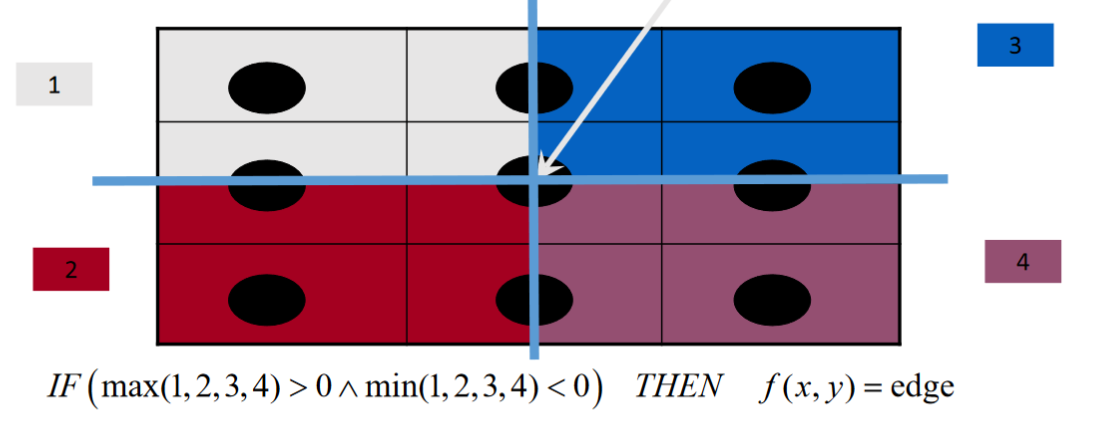
原图像 Laplacian卷积 Laplacian卷积 +阈值

## 3.5 Laplacian算子改良方法（Marr-Hildreth算子）

之前提过Laplacian 算子对噪声敏感(sensitive)的问题，噪声响应比一阶算子还大。一个改良方法是对Laplacian算子和Gaussian模板结合起来做一个平滑(smooth)处理，可以得到Laplacian of Gaussian（LOG）算子



Marr-Hildreth算子在LOG上继续发展，其中碰到了过零点检测（Zero crossing detection）问题，解决方法是查看四个象限。假如图像是一个3X3区域，把图像分割成4个区域，区域1表示左上的4个像素点，以此类推。



如果满足此公式说明中心处一定有一个过零点

优缺点分析

优点：

Marr-Hildreth算子的一个优点是能给出封闭的边缘边界 (ability to provide closed edge boards), 然而对应的一阶Canny算子则不能。

另外一个优点是能避免滞后阈值的递归计算(avoids the recursion associated with hysteresis thresholding)，从而节约了大容量堆栈(massive stack size)

缺点:

Marr-Hildreth算子的实现和真正的LOG算子实现有有一些不同

（备注：具体哪里不同，书上也没说，只说了在1985年发现的）

(Unfortunately, the implementation appeared to be different from the original LOG operator)

LOG算子大小很难选择适当，在噪声过滤中所需药的算子大小与目标特征的大小之间需要折中

# Finding Shapes (Hough Transform霍夫变换)

Finding Shapes是高层次特征处理 (High-level feature extraction)

霍夫变换(Hough Transform) 霍夫变换是图像处理中从图像中识别几何形状的基本方法之一。主要用来从图像中分离出具有某种相同特征的几何形状（如，直线，圆等。

霍夫的主要优势在于可以得到与模板匹配(Template matching)相同的结果，但是速度更快。因为它是基于证据收集(Evidence-gathering) 方法，通过模板匹配过程的重新描述实现的。

它通过一种投票(vote)算法检测具有特定形状的物体。该过程在一个参数空间中通过计算累计结果的局部最大值得到一个符合该特定形状的集合作为霍夫变换结果.

霍夫变换的优点：计算资源少

霍夫变换的缺点：需要大量存储空间

Template matching

1. Construct a template containing the shape trying to find
2. Centre the template on an image point and count how many points in the template matched those in the image
3. Repeat for entire image. Maximum count is the best match.

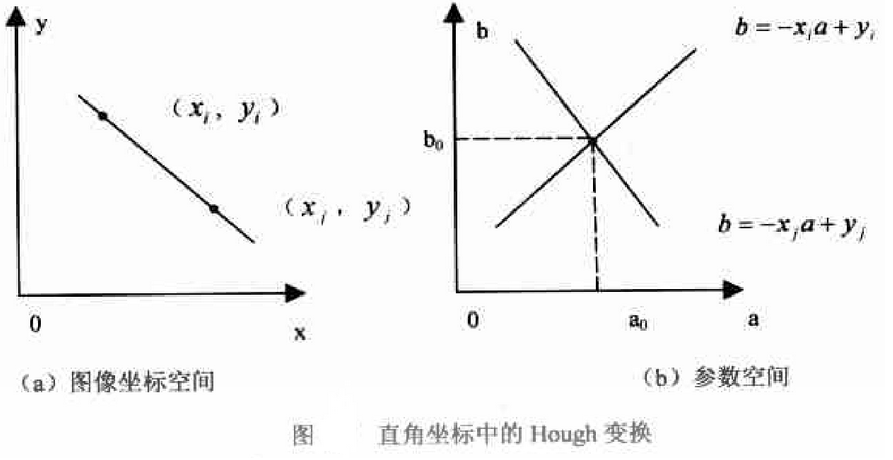
## 直线霍夫变换

在笛卡尔坐标系（Cartesian coordinates）中， 直线方程是

假设a,b已知，那么直线就确定。 任意选取两个点(xi,yi)和（xj,yj）

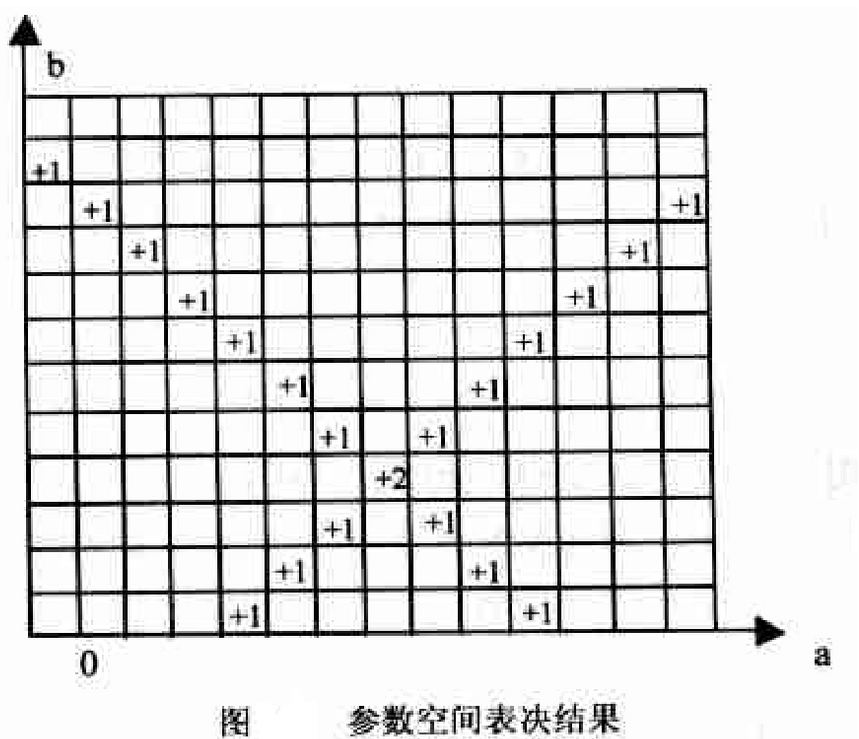
霍夫变换是将变量(x,y)和参数(a,b)进行反转，此时将x,y看成参数，a,b看成变量，反解得到 

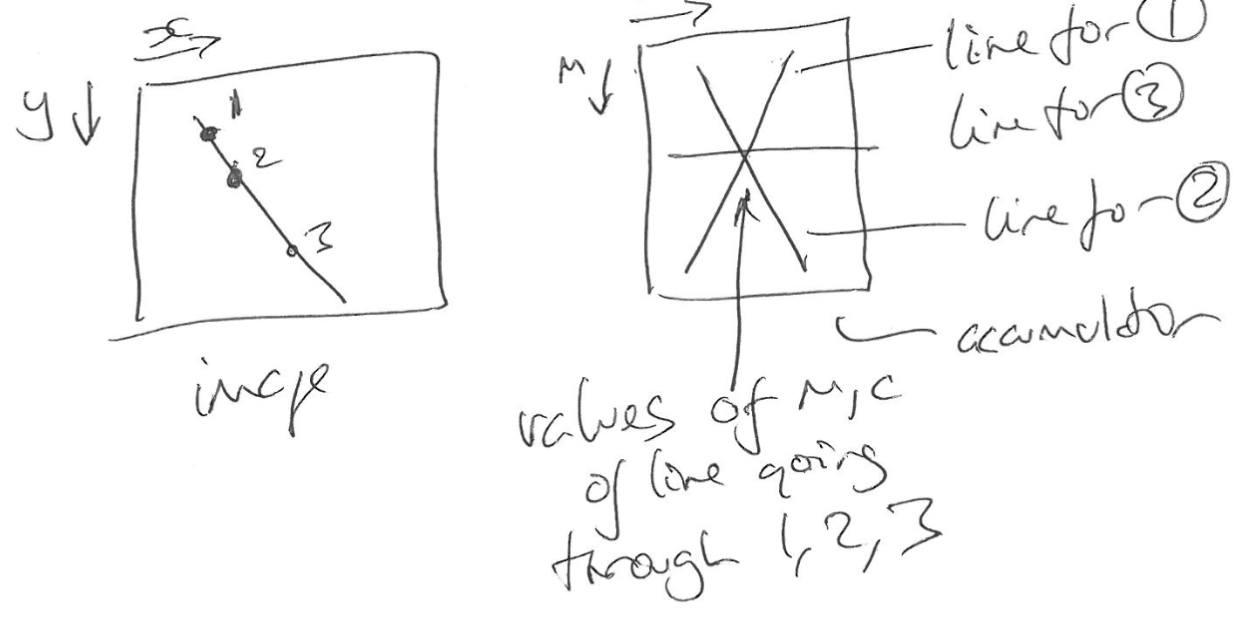
这个变换就是直角坐标中对于(xi,yi)点的Hough变换。由于我们之前知道这个直线过点(xi,yi)和（xj,yj），在a-b参数空间坐标系中 , x和y作为斜率和截距为已知，正好对应两条直线方程，如下图所示。



这些直线都相交于点(a0,b0),而a0、b0就是图像坐标空间x−y中点(xi,yi)和点(xj,yj)所确定的直线的参数。

反之，在参数空间相交于同一点的所有直线，在图像坐标空间都有共线的点与之对应。根据这个特性，给定图像坐标空间的一些边缘点，就可以通过Hough变换确定连接这些点的直线方程。再利用累加器(accumulator)进行简单计数，将直线的提取问题转换为爱累加器空间(既a-b参数空间)中定位最大值的问题。





## Foot-of-normal （极坐标直线霍夫检测）

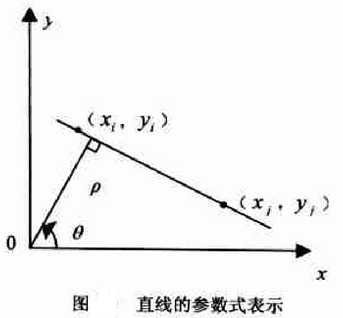
直线霍夫变换中，使用笛卡尔坐标表示直线时，斜率可以为无穷大，无法表示出来(a,b tend to infinity)（备注：当xi=0时，b为yi，可以自由取值到无穷）

因此，解决办法是用极坐标参数空间来表示

极坐标中用如下参数方程表示一条直线



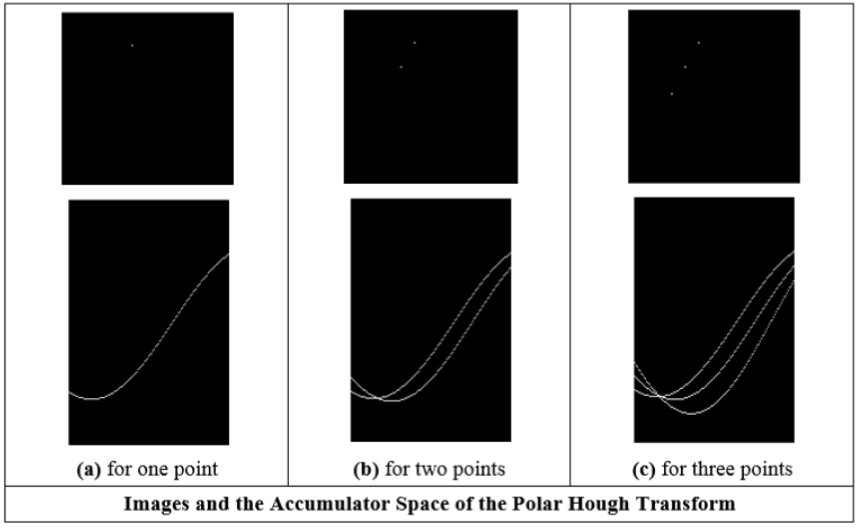
其中，ρ代表直线到原点的垂直距离，θ代表x轴到直线垂线的角度，取值范围为±90∘，如图所示



与直角坐标类似，极坐标中的霍夫变换也将图像坐标空间中的点变换到参数空间中。   
在极坐标表示下，图像坐标空间中共线的点变换到参数空间中后，在参数空间都相交于同一点，此时所得到的ρ、θ即为所求的直线的极坐标参数。

具体计算时，与直角坐标类似，也要在参数空间中建立一个二维数组累加器A，按照正弦波方式投票(Sinusoidal manner)。只是取值范围不同。对于一副大小为D×D的图像，通常ρ的取值范围为[−2√D/2,2√D/2] ,θ的取值范围为[−90∘,90∘]。计算方法与直角坐标系中累加器的计算方法相同，最后得到最大的A所对应的(ρ,θ)

与直角坐标不同的是，用极坐标表示时，可以在累加器空间画出不同的点(loci)，点被映射(map)到累加器空间的曲线(curves)



## 圆的霍夫变换

同理，在笛卡尔坐标系下知道圆的方程为 

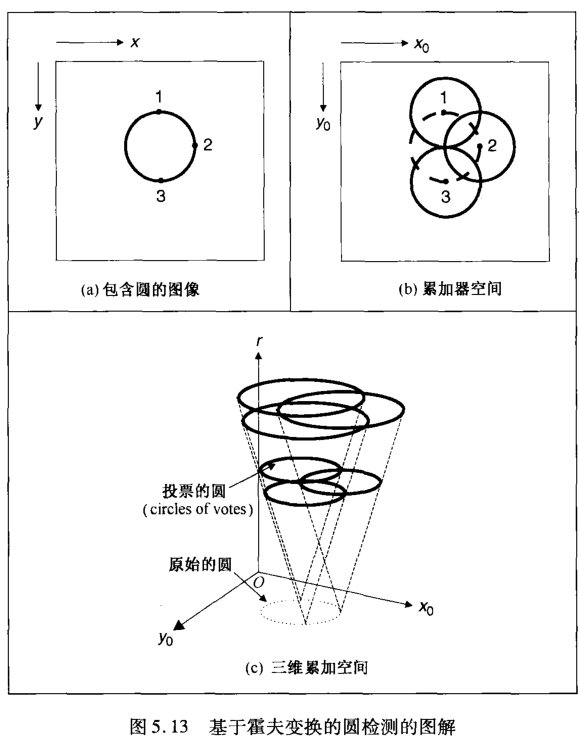
该方程定义了以原点(x0,y0)为圆心，r为半径的点的轨迹，这个圆过(x1,y1), (x2,y2),(x3,y3)这三个点

反解该方程，霍夫变换将x,y看成参数，x0,y0,r看成变量。 很显然这里有三个变量，所以霍夫变换后是在三维空间中的

此时，霍夫变换的图像就是以(x1,y1), (x2,y2),(x3,y3)这三个点为圆心，半径为r的一系列圆的集合。

三维累加器空间中，边缘点映射了投票椎体。

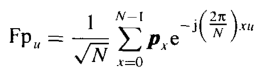
(In the accumulator space, edge points map to a cone of votes)

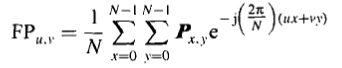


枚举出足够多数量后的圆集合后，累加器空间中最大值就是原始图像的圆的参数

# 离散傅里叶变换(Discrete Fourier Transform)

离散傅里叶变换是连续傅里叶变换的离散模拟，由于图像处理关心的是采样后的数据，而这个步骤需要离散傅里叶变换来完成。定义：

一维：

二维：

离散傅里叶变换可以将图像从时域(time domain)变成频域(frequency domain)

在频域中，低频(low-frequency)部分带有大量信息（much of the information is carried），因为大多数频谱分量集中在这里(most of the spectral components concentrate)。反映了图像的大致轮廓(contour).

高频(high-frequency)部分是反映亮度(intensity)上的变换，既对应了图像的细节(detail)

傅里叶变换的整体效果通过高分辨率图像的应用得到显示

(The full effect of the Fourier transform is shown by application to an image of much higher resolution)

加速，滤波