

边缘检测算法

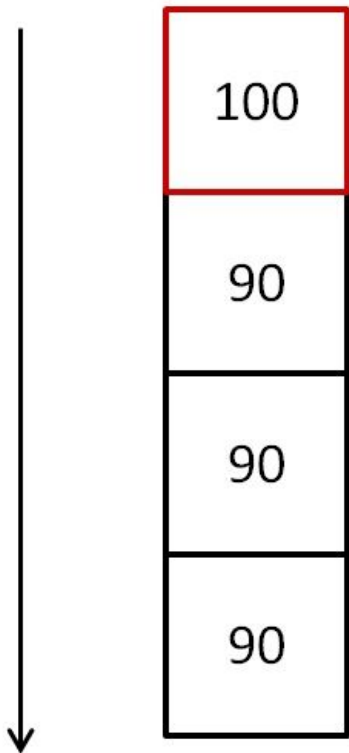
图像边缘检测的原理是检测出图像中所有灰度值变化较大的点，而且这些点连接起来就构成了若干线条，这些线条就可以称为图像的边缘。常见的边缘检测算法包括Soble边缘检测，拉普拉斯边缘检测和Canny边缘检测。边缘检测本质上就是一种滤波算法。

图像的滤波一般是基于灰度图进行的，因此图像此时是二维的，因此我们在看一下二维函数的微分，即偏微分方程：

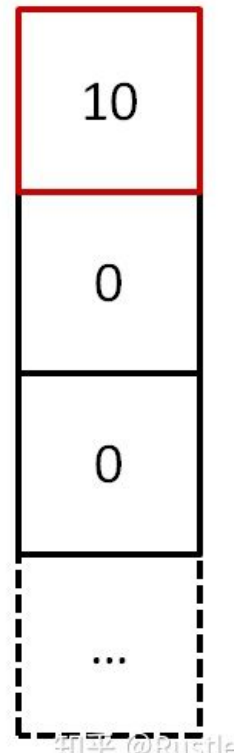
$$\frac{\partial f(x, y)}{\partial x} = \lim_{\epsilon \rightarrow 0} \frac{f(x + \epsilon, y) - f(x, y)}{\epsilon}$$

$$\frac{\partial f(x, y)}{\partial y} = \lim_{\epsilon \rightarrow 0} \frac{f(x, y + \epsilon) - f(x, y)}{\epsilon}$$

原图像
某像素X方向



该像素X方向
梯度



Canny边缘检测

- 1. 图像降噪：我们第一步就是想先去噪声，因为噪声就是灰度变化很大的地方，所以容易被识别为伪边缘。
- 2. 计算图像梯度：计算图像梯度能够得到图像的边缘，因为梯度是灰度变化明显的地方，而边缘也是灰度变化明显的地方。
- 3. 非极大值抑制
- 4. 阈值筛选

Sobel边缘检测算子

Sobel算子有两个，一个是检测水平边缘的；另一个是检测垂直边缘的。它对于像素的位置的影响做了加权，可以降低边缘模糊程度，因此效果更好。

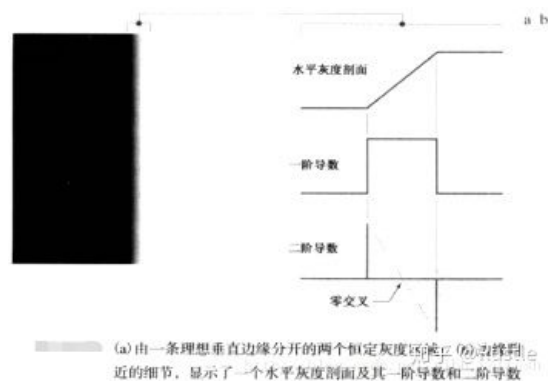
$$s_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, s_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}, K = \begin{bmatrix} a_0 & a_1 & a_2 \\ a_7 & [i, j] & a_3 \\ a_6 & a_5 & a_4 \end{bmatrix}$$

Laplacian算子

Laplacian算子是一个实现二阶微分的模板。二阶微分可以利用两个相邻一阶微分的差值来近似。Laplacian算子法对噪声比较敏感。基于拉普拉斯变换的图像增强已成为图像锐化处理的基本工具。

图像锐化(image sharpening)是补偿图像的轮廓，增强图像的边缘及灰度跳变的部分，使图像变得清晰，分为空间域处理和频域处理两类。

Marr-Hildreth算子



如图所示，左边表示的是一副灰度图像，从左到右从黑色(0)慢慢变为白色(255)，现在我们来查看它的水平灰度剖面，灰度值从小到大平稳上升，其一阶导数表示为在上升区域为不变的值，其中得到的信息是图像灰度值是平稳过渡的，即梯度值相等，接下来在将其求二次导数，得到的图像为在开始过渡的起点为正数，其值为一阶导数在此点的梯度值，结束点也和起点一样，现在重点来了，我们将这两点连起来得到一个与X轴的交叉点，这一点就是我们所认为的边缘点，这就是二阶导数应用在边缘检测领域的奇妙之处。

计算零交叉(Zero crossing)

零交叉的实现较为简单，由于零交叉点意味着至少两个相邻的像素点的像素值异号，一共有四种需要检测的情况：左右，上下，两个对角，其中如果滤波后的图像g(x, y)的任意像素p处的四种情况其中一组的差值的绝对值超过了设定的阈值，则我们可以称p为一个零交叉像素。

Roberts边缘算子

Roberts边缘算子是一个2x2的模板，采用的是对角方向相邻的两个像素之差。

Prewitt边缘检测算子

Prewitt利用周围邻域8个点的灰度值来估计中心的梯度。

补充

- 1.滤波器的尺寸应该是奇数，这样才有一个中心点可进行赋值操作，常见的滤波器或卷积核(Conv kernel)有3x3、5x5等，因此也有了半径的概念，例如5x5的卷积核的半径为2
- 2.滤波器中所有元素之和应为0，这一限制条件是保证滤波前后图像总体灰度值不变
- 3.Roberts算子、Sobel算子、Prewitt算子运算速率高，对噪声也有一定抑制作用，但检测出的边缘质量不高，如边缘较粗、定位不准、间断点多
- 4.Canny算子不容易受噪声干扰，得到的边缘精细且准确，缺点就是运算代价较高，运行于实时图像处理较困难，适用于高精度要求的应用
- 5.Marr-Hildreth算子边缘检测效果相对较优，但对于噪声比较敏感（因其二阶运算的性质）