Sobel 算子的数学基础

彭真明（10/06/2018）

图像处理及机器视觉中的 Sobel 算子，是以它的提出者 Irwin Sobel 的名字命名。因为这个算子没有正式发表过，仅仅是在一个非正式的博士生课题讨论组会上与 Gary Feldman（当时的一个在读研究生，也是 Sobel 的朋友）一起提出，I. Sobel 还曾专门撰文申明，建议把 Sobel 算子的命名改成“Sobel-Feldman”算子。

对于这个看似简单却沿用了几十年的算子，包括国际上最权威的冈萨雷斯《数字图像处理》教材在内，很多教科书并没有对算子的原理和数学基础做具体描述。

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

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

图 1 Sobel 算子（模板）

Sobel 算子考虑了水平、垂直和 2 个对角共计 4 个方向对的梯度加权求和，是一个 3x3 各向异性的梯度算子。另外，大部分教科书停留在对算子的直观解释上，一般描述为隔行或隔列的差分运算，然后做中心像素位置的 2 倍加权。实际上，并不是简单的隔行/列的差分运算，中心像素位置并未参与运算。

Sobel 算子具有严格的数学基础，主要关键点在于：

1）笛卡尔网格（Cartesian grid）

2）前向差分（Forward-difference）

3）距离反比的 4 方向对梯度加权

4）城市距离（city-block distance）

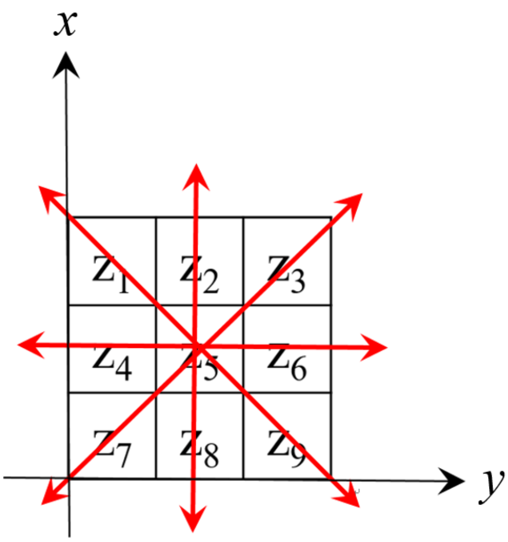
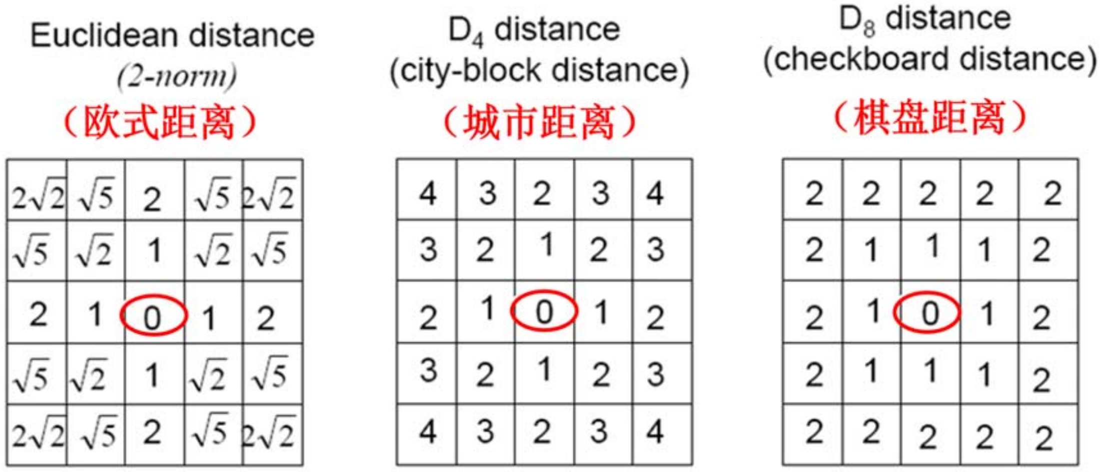


图 2 像素 N_8(p)邻域及 Cartesian 网格



Euclidean distance
(2-norm)
（欧式距离）

$2\sqrt{2}$	$\sqrt{5}$	2	$\sqrt{5}$	$2\sqrt{2}$
$\sqrt{5}$	$\sqrt{2}$	1	$\sqrt{2}$	$\sqrt{5}$
2	1	0	1	2
$\sqrt{5}$	$\sqrt{2}$	1	$\sqrt{2}$	$\sqrt{5}$
$2\sqrt{2}$	$\sqrt{5}$	2	$\sqrt{5}$	$2\sqrt{2}$

D_4 distance
(city-block distance)
（城市距离）

4	3	2	3	4
3	2	1	2	3
2	1	0	1	2
3	2	1	2	3
4	3	2	3	4

D_8 distance
(checkboard distance)
（棋盘距离）

2	2	2	2	2
2	1	1	1	2
2	1	0	1	2
2	1	1	1	2
2	2	2	2	2

图 2 三种邻域像素距离定义

定义一个给定邻域方向梯度矢量 **g** 的幅度为

|**g**| = <像素灰度差分>/<相邻像素的距离>

Sobel 采用的像素距离是一种城市距离（city-block），而并非通常的欧式距离。因此，对角相邻像素之间的距离值为 2。

矢量'**g**'的方向可以通过中心像素“z5” 相关邻域的单位适量给出，这里的邻域是对称出现的，四个方向即： (z1, z9) ，(z2, z8)， (z3, z7)， (z6, z4). 沿着 4 个方向求其梯度和，可以给出当前像素的梯度估计，则有

$$\mathbf{G} = (z3-z7)/4 * [1, 1] + (z1-z9)/4 * [-1, 1] + (z2-z8)/2 * [0, 1] + (z6-z4)/2 * [1, 0]$$

式中的系数 1/4,1/2 为距离反比权重。

上式展开后，有

$$G = [(z3- z7- z1+ z9)/4 + (z6-z4)/2, (z3- z7+ z1- z9)/4 + (z2-z8)/2]$$

注意，上述公式里并没有求平方根。如果要求数字上的精确度，上式需要除以 4 得到平均梯度值。然而，一些典型的运算都是针对数值较小的整数的定点运算，除法会丢失低阶的重要字节, 更方便的是把向量乘于 4，而不是除于 4，以保留低阶字节。因此，计算出的估计值比平均梯度数值上扩大了 16 倍。公式变为：

$$\begin{aligned} G' &= 4*G \\ &= [z3- z7- z1+ z9 + 2*(z6-z4), z3- z7+ z1- z9 + 2*(z2-z8)] \\ &= [z3+2*z6+z9 - z1-2*z4-z7, z1+2*z2+z3-z7 - 2*z8-z9] \end{aligned}$$

按 z-y 方向，可分别写成：

$$G'x = (z3+2*z6+z9) - (z1+2*z4+z7)$$

$$G'y = (z1+2*z2+z3) - (z7 +2*z8+z9)$$

以上式子，就与我们教科书上的表达式完全一致了，也很容易得出如图 1 所示的两方向 Sobel 算子模板。

在邻域像素距离模型上，如果改用欧式距离，就不难得到 Frei 和 Chen （1991）的边缘检测器了。

$$G_1 = \frac{1}{2\sqrt{2}}\begin{bmatrix} -1 & -\sqrt{2} & -1 \\ 0 & 0 & 0 \\ 1 & \sqrt{2} & 1 \end{bmatrix} \quad G_2 = \frac{1}{2\sqrt{2}}\begin{bmatrix} 1 & 0 & -1 \\ \sqrt{2} & 0 & -\sqrt{2} \\ 1 & 0 & -1 \end{bmatrix}$$

另外，教科书上的一些经典滤波器，其实也都有它的计算模型，而不是人为的随意构建，如拉普拉斯滤波器，高斯滤波器，LoG 滤波器等。作为教师，应该对这些算子或滤波器的数学基础和模型原理有很深入的理解，不能给学生打马虎。

[1] Sobel I, Feldman G. A 3x3 isotropic gradient operator for image processing[C]. a talk at the Stanford Artificial Project in 1968: 271-272.

[2]https://www.researchgate.net/publication/239398674_An_Isotropic_3_3_Image_Gradient_Operator

[3] http://rastergrid.com/blog/2011/01/frei-chen-edge-detector/

[4] 彭真明，作者 40 年后谈 Sobel 算子的由来和定义，http://blog.sciencenet.cn/blog-425437-776050.html