

一种二值图像的边缘提取算法实现

李哲丰

(中国地质大学(武汉) 机械与电子信息学院, 湖北武汉 430074)

[摘要] 边缘提取属于图像理解的低层处理阶段,即提取被选定物体区域的外边轮廓。本文主要介绍了一种新的二值图像边缘提取算法。通过分析区域边缘像素的特征,归纳出边缘提取条件,该算法逐行扫描像素区域,并分析该像素区域特征可以无损失地提取出所有边缘像素点。

[关键词] 二值图像;边缘提取;邻域matlab算法

图像边缘检测是图像处理中的一个重要步骤,是其他图像处理的必要基础,因此图像边缘检测算法的研究具有极其重要的地位。边缘检测的实现有许多方法,比如跟踪法、基于模板的算法等等。

1 几种经典图像边缘检测算法

1.1 Sobel算子

算子如下:

$$h_1 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad h_2 = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

其差分计算式定义如下:

$$f_x = \begin{vmatrix} f(x+1, y-1) + 2f(x+1, y) + f(x+1, y+1) - f(x-1, y-1) - 2f(x-1, y) - f(x-1, y+1) \\ f(x-1, y+1) + 2f(x, y+1) + f(x+1, y+1) - f(x-1, y-1) - 2f(x, y-1) - f(x+1, y-1) \end{vmatrix}$$

Sobel算子是利用像素的上下左右邻域的灰度加权算法、根据在边缘点处达到极值的原理进行图像的边缘检测。该算子不但检测效果好,而且具有很好的噪声平滑作用,同时也包含比较准确的边缘方向信息。

1.2 Canny算法

在Canny算法中,主要对水平方向与垂直方向处理,使用了高斯滤波及其一阶差分,然后利用线性插值的方法来计算得到边缘的梯度以及角度。但是边缘平面并不一定是一个线性平面,所以插值结果通常有较大误差。

1.3 边缘跟踪、标记算法

对一幅图片顺序进行光栅扫描,就会发现未分配标记的1像素点(定义二值图像为0或1像素,其中1像素为目标像素,下同)。对这个像素点分配给它还没使用过的标号,对位于这个像素点8-邻域内的1像素点赋予相同的标号,然后对位于其8-邻域的1像素点也赋予相同的标号。反复进行这一处理,直到应该标号的1像素点已经标记完全时,对其1像素链分配相同标号的操作结束。继续对图像进行扫描,如果发现没有分配标号的1像素点就赋给新的标号,进行同上处理,否则处理结束。

2 本文算法介绍

通过对二值图像的观察和分析,对于任意一像素点,假如如是边界点,那么该点的像素值为1,并且在其邻域内至少存在一个0像素点。由排列组合知识可以得到16种可能情况,但其中有两种不是边缘情况(四像素点为全0或全1值),其余14种情况如下:

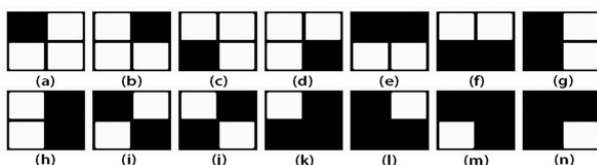


图1

其中黑色区域代表1像素值,白色区域代表0像素值,因此图像所有的边缘扫描结果都包含在了上面14种情况中。下面按两种边缘定义方法来确定边缘坐标。

先介绍邻域与邻接,像素P(如图2(a)所示)上下左右的4个

像素{P0、P2、P4、P6}称为像素的4-邻域,如图2(b)所示。互为4-邻域的两个像素叫做4-邻接(或4-连通),图2(a)中P和P0、P0和P1等均为4-邻接。

像素P上下左右的4个像素和4个对角线像素即P0~P7称为像素P的8-邻域,如图2(c)所示。互为8-邻域的两个像素叫做8-邻域(或8-连通)。图2(a)中P和P1、P0和P2等均为8-邻接。

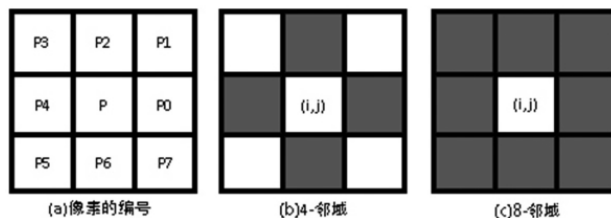


图2

当以4-邻接为基础进行边缘提取时,根据上面4-邻接的定义可得某一像素点P,对于4-邻接边缘检测,边缘只能是横平竖直地进行连续标记。

同4-邻接分析,对于图1中的14种情况,可以知道所有1像素点都是相互关联,满足8-邻接定义,但是8-邻接边缘有一种情况值得我们注意,如下图3,其中(a)为各像素点的编号,(b)为设定区域范围,根据8-邻接边缘定义,可得其边缘像素点应为(c)所示P1、P、P7三点,而没有P0和P6点,因此在逐行扫描像素区域时,当扫描到P2 P P1 P0区域以及P4 P5 P P6区域,对应于图1中(k)情况,其(2,2)像素点不属于边缘区域,因此在算法实现中需要排除,同理①、(m)、(n)也属于同样情况,需要排除一个不符合条件的像素点。

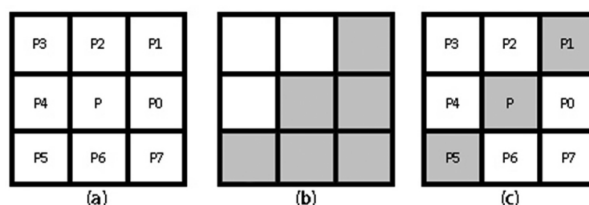


图3

3 具体算法实现

对于任一点像素点,其值非1即0,因此当扫描对象为一个2×2像素区域时,四个像素值的和有0、1、2、3、4五种情况,当像素和为0时,说明四像素点为全黑点,当像素和为4时,说明四像素点为全白点,而这都不属于图1中14种情况之一,当像素和为1或2或3时,说明四点中既有0像素点也有1像素点,同时也对应着图1中14种情况中的一种,即我们需要处理的像素区域点集。

因此在逐行扫描时只需筛选出像素和等于1或2或3的像素区域,当为4-邻接边缘描述时,只需标记出所有满足条件的2×2像素区域中的1像素点即可得到图中所有的边缘像素点(可能有像素点被重复描述,但标记的内容是一样的,因此也是正确的)。

当为8-邻接边缘描述时,根据上文已得结论可知:在像素区域四像素点和为3时,需要排除一像素点,下面将分析排除该点的实现过程。

(下转第37页)

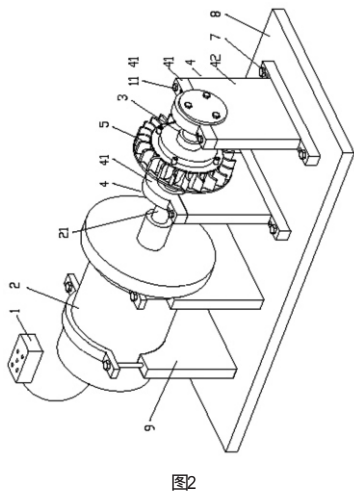


图2

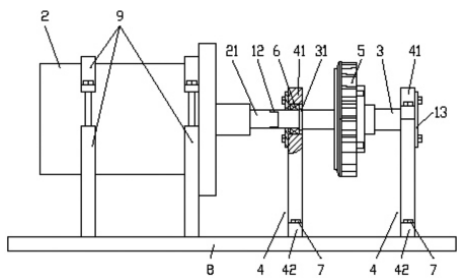


图3

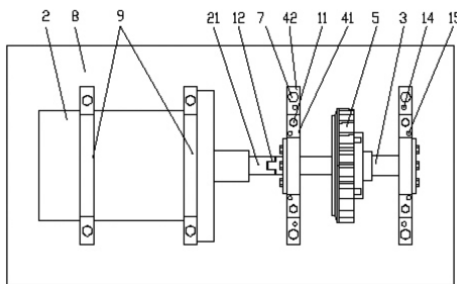


图4

作者简介：董明，1977 生，男，汉族，西南大学机械设计制造学士，重庆大学工商管理硕士。

(上接第 23 页)

由 matlab 基本知识知，对于一个矩阵 M ，当以一维形式表达矩阵时为 $M(1) - M(4)$ ，与二维表达形式的对应关系为：

$M(1) \leftrightarrow M(1,1)$

$M(2) \leftrightarrow M(2,1)$

$M(3) \leftrightarrow M(1,2)$

$M(4) \leftrightarrow M(2,2)$

$M(1)$	$M(3)$
$M(2)$	$M(4)$

因此只要以一维形式进行标示，再对标示的坐标值进行求和即可区分得到图 1 中 (k) - (n) 四种情况中的一种，例如当为 (k) 情况时， $M(2)$ 、 $M(3)$ 、 $M(4)$ 为标示位，所以坐标值和为 $2+3+4=9$ ，同理可得 (l)、(m)、(n) 三种情况坐标值和分别等于 8、7、6，由此可以区分同类型 (k) - (n) 四种不同情况，并进行相应正确的标记。

4 算法实现结果

本文使用 matlab7.1 进行程序编写。以二值化处理后的 Lena 图像（如图 4 (a)）作为输入数据，分别基于 4- 邻接、8- 邻接进行边缘描述，测试结果如下图：



图4

5 结语

本文算法相比于几种传统的边缘检测算法，具有 100% 标记原图边缘信息的特点，并且算法原理简单易懂，但是正因为可以完全标记原图所有边缘信息，所以假若原图的边缘信息不完全，则本算法不具有边缘信息修复输出的功能，因此在实际应用中可以与其他算法综合使用，以达到更好的边缘描述效果。

作者简介：李哲丰，1990 年生，男，湖南省湘潭市人，研究方向为图像处理与图形学。

[参考文献]

[1] 郭文强,侯勇严. 数字图像处理[M]. 西安:西安电子科技大学出版社,2009.
[2] Rafael C.Gonzalez,Richard E.Woods. 数字图像处理(第二版)[M]. 北京:电子工业出版社,2007.
[3] 吴雪刚. 二值图像边缘提取的新算法[J]. 科技创新导报,2007.
[4] 王福生,齐国清. 二值图像中目标物体轮廓的边界跟踪算法[J]. 大连海事大学学报,200