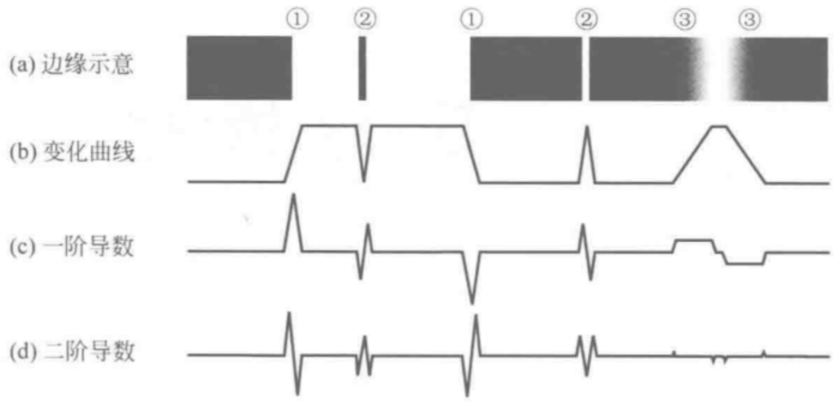
《数字图像处理》期末速通教程

7. 图像锐化

7.1 图像边缘与图像锐化

[**定义 7.1.1**] **边缘**是图像中亮度突变的区域, 分为三种类型: ① 突变型边缘; ② 细线型边缘; ③ 渐变型边缘. 将三种边缘放在同一图像中, 绘制灰度变化曲线和曲线的一阶、二阶导数曲线, 如下图所示.



(1) 突变型边缘: 位于两个灰度值不同的区域间.

曲线特点: 灰度曲线阶跃, 一阶导数极值点, 二阶导数过零点.

(2) 细线型边缘:

曲线特点: 灰度曲线极值点, 一阶导数过零点, 二阶导数极值点.

(3) 渐变型边缘:

曲线特点: 灰度变化缓慢, 无明确的边界点.

综上, 可用微分检测图像边缘.

[**定义 7.1.2**] **图像锐化**加强图像中景物的边缘和轮廓, 突出图像中的细节或增强被模糊的细节.

[**注**] 滤波是积分的过程, 使图像模糊; 锐化是微分的过程, 能增加细节信息, 使图像清晰.

7.2 一阶微分算子

7.2.1 梯度算子

[**定义 7.2.1.1**] 使用**梯度算子**时, 图像 $f(x,y)$ 的**梯度图像**定义为

$$g(x,y)=|f(x+1,y)-f(x,y)|+|f(x,y+1)-f(x,y)|\ \ .$$

特别地, 模板罩不住的像素点的像素值赋背景值 0 .

[**注 1**] 图像锐化的本质: 原图像和梯度图像相加, 以增强图中的变化.

[**注 2**] 边缘检测中还需进一步判断梯度图像的极值点, 一般通过阈值化实现: 设定一阈值, 梯度大于该阈值的点的梯度设为 1 , 表示边缘点; 梯度小于阈值的点的梯度设为 0 , 表示非边缘点.

边缘检测的效果与阈值有关:

(1) 阈值越低, 能检测出的边线越多, 但结果更易受图像噪声影响.

(2) 阈值越高, 能检测出的边线越少, 但可能遗失较弱的边线.

实际边缘检测前可先滤波, 降低噪声的影响, 或采用不同的方法选取合适的阈值.

7.2.2 Robert 算子

[**定义 7.2.2.1**] 使用 **Robert 算子**时, 图像 $f(x,y)$ 的**梯度图像**定义为

$$g(x,y)=|f(x,y)-f(x+1,y+1)|+|f(x+1,y)-f(x,y+1)|\ \ .$$

特别地, 模板罩不住的像素点的像素值赋背景值 0 .

7.2.3 Sobel 算子

[**定义 7.2.3.1**] 使用 **Sobel 算子**时, 图像 $f(x,y)$ 的**梯度图像**定义为

$$\begin{cases}S_x=|f(x-1,y+1)+2f(x,y+1)+f(x+1,y+1)| \\ \qquad\quad -|f(x-1,y-1)+2f(x,y-1)+f(x+1,y-1)| \\ S_y=|f(x+1,y-1)+2f(x+1,y)+f(x+1,y+1)| \\ \qquad\quad -|f(x-1,y-1)+2f(x-1,y)+f(x-1,y+1)| \\ g(x,y)=|S_x|+|S_y|\end{cases}\ \ .$$

特别地, 模板罩不住的像素点的像素值赋背景值 0 .

[**注 1**] Sobel 算子的优点:

(1) 引入平均元素, 对图像中的随机噪声有一定的平滑作用.

(2) 相邻两行或两列求差分, 边缘两侧的元素得到增强, 边缘显得粗而亮.

[**注 2**] Sobel 算子的模板通过旋转可扩展为 8 个模板.

两种算子视觉效果区别不大, 但扩展算子检测边缘时方向信息更丰富, 在需要边缘信息的情况下, 扩展算子应用更广泛.

[**例 7.2.3.1**] 求图像 $f=\begin{bmatrix}3&3&3&3&3\\3&7&7&7&3\\3&7&7&7&3\\3&7&7&7&3\\3&3&3&3&3\end{bmatrix}$ 的梯度图像 g , 分别采用:

[1] 梯度算子.

[2] Robert 算子.

[3] Sobel 算子.

[**解**]

[1] $g(0,0)=|f(1,0)-f(0,0)|+|f(0,1)-f(0,0)|=|3-3|+|3-3|=0$.

$g(0,1)=|f(1,1)-f(0,1)|+|f(0,2)-f(0,1)|=|7-3|+|3-3|=4$.

类似地求其它点处的像素值. 梯度图像 $g=\begin{bmatrix}0&4&4&4&0\\4&0&0&4&0\\4&0&0&4&0\\4&4&4&8&0\\0&0&0&0&0\end{bmatrix}$.

[2] $g(0,0)=|f(0,0)-f(1,1)|+|f(1,0)-f(0,1)|=|3-7|+|3-3|=4$.

类似地求其它点处的像素值. 梯度图像 $g=\begin{bmatrix}4&8&8&4&0\\8&0&0&8&0\\8&0&0&8&0\\4&8&8&4&0\\0&0&0&0&0\end{bmatrix}$.

[3] 以点 $(0,0)$ 为中心时, 模板罩不住, 则 $g(0,0)=0$.

对点 $(1,1)$,

$$\begin{cases}S_x=|f(0,2)+2f(1,2)+f(2,2)|-|f(0,0)+2f(1,0)+f(2,0)|=12\\S_y=|f(2,0)+2f(2,1)+f(2,2)|-|f(0,0)+2f(0,1)+f(0,2)|=12\\g(1,1)=|S_x|+|S_y|=24\end{cases}.$$

类似地求其它点处的像素值. 梯度图像 $g=\begin{bmatrix}0&0&0&0&0\\0&24&16&24&0\\0&16&0&16&0\\0&24&16&24&0\\0&0&0&0&0\end{bmatrix}$.

7.3 二阶微分算子

[**定义 7.3.1**]

(1) **Laplace 算子**是一种二阶微分算子, 其模板 $H_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$ 或 $H_1 = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$.

(2) 使用 Laplace 算子作图像锐化的模板 $H = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$.

特别地, 模板罩不住的像素点的像素值赋背景值 0 .

[**注 1**] 使用 Laplace 算子时可能得到负的像素值, 有如下两种处理方法:

(1) 取绝对值, 得到梯度图像.

(2) 整体加一个正整数(最小像素值的绝对值), 得到类似浮雕的效果.

[**注 2**] 一、二阶微分算子在提取图像细节信息上的异同:

(1) 同:

① 都用于图像锐化和图像增强, 通过计算梯度提取图像的高频信息.

② 都结合其它滤波器改善效果, 减少噪声影响.

(2) 异:

① 一阶微分算子主要用于边缘检测; 二阶微分算子主要用于增强图像细节和纹理.

② 一阶微分算子有方向性, 可检测特定方向的边缘; 二阶微分算子通常无方向性.

③ 二阶微分算子对噪声更敏感, 需更多的预处理以抑制噪声的影响.

[**例 7.3.1**] 对图像 $f = \begin{bmatrix} 3 & 3 & 3 & 3 & 3 \\ 3 & 7 & 7 & 7 & 3 \\ 3 & 7 & 7 & 7 & 3 \\ 3 & 7 & 7 & 7 & 3 \\ 3 & 3 & 3 & 3 & 3 \end{bmatrix}$ 使用 Laplace 算子, 取模板 $H_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$.

[**解**] 以点 $(0,0)$ 为中心时, 模板罩不住, 则 $g(0,0)=0$.

$g(1,1) = \nabla^2 f = f(2,1) + f(0,1) + f(1,2) + f(1,0) - 4f(1,1) = -8$.

类似地求其它点处的像素值. 结果图像 $g = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -8 & -4 & -8 & 0 \\ 0 & -4 & 0 & -4 & 0 \\ 0 & -8 & -4 & -8 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$.