《数字图形处理》实验报告

实验 3：边缘检测

姓名　齐琪格

学号　6319000163

专业　智能与计算学部

班级　计科三班

1&2：数学证明

1. Give a math deduction to the general form of "discrete Laplacian operator"

连续函数下的 Laplacian 算子为

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

导数的定义为 $f'(x) = \lim_{\Delta x \to 0} \frac{f(x+\Delta x) - f(x)}{\Delta x}$

在图像中, Δx 的最小取值为 1

$$f'(x) = f(x+1) - f(x)$$

同理, 在二维情况下

$$\frac{\partial f}{\partial x} = f(x+1, y) - f(x, y)$$

$$\frac{\partial f}{\partial y} = f(x, y+1) - f(x, y)$$

$$\frac{\partial^2 f}{\partial x^2} = \frac{\partial f(x+1, y)}{\partial x} - \frac{\partial f(x, y)}{\partial x}$$

$$= f(x+2, y) - f(x+1, y) - f(x+1, y) + f(x, y)$$

$$= f(x+2, y) + f(x, y) - 2f(x+1, y)$$

令 $x = x-1$

$$= f(x+1, y) + f(x-1, y) - 2f(x, y)$$

同理

$$\frac{\partial^2 f}{\partial y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y)$$

$$\nabla^2 f = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4f(x, y)$$

2. Give a math deduction to the form of "Laplacian of Gaussian"

高斯函数为 $G_\sigma(x,y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\left(-\frac{x^2+y^2}{2\sigma}\right)}$

先进行高斯变换后再使用 ∇ 算子

$$\nabla^2 [G_\sigma(x,y) * f(x,y)] = [\nabla^2 G_\sigma(x,y)] * f(x,y)$$
$$= LoG * f(x,y)$$

在卷积中，有 $\frac{d(h(t) * f(t))}{dt} = f(t) * \frac{dh(t)}{dt}$

LoG中可以先对Gauss求导

$$\frac{\partial G_\sigma(x,y)}{\partial x} = -\frac{x}{\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$$

求二阶导数得

$$\frac{\partial^2 G_\sigma(x,y)}{\partial^2 x} = \frac{x^2-\sigma^2}{\sigma^4} e^{-(x^2+y^2)/2\sigma^2}$$

$$\begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & 1 & 0 \\ 1 & 2 & -16 & 2 & 1 \\ 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

5x5 LoG sample

同理

$$\frac{\partial^2 G_\sigma(x,y)}{\partial^2 y} = \frac{y^2-\sigma^2}{\sigma^4} e^{-(x^2+y^2)/2\sigma^2}$$

综上所述 LoG 可以表示为

$$LoG = \frac{x^2+y^2-2\sigma^2}{\sigma^4} e^{-(x^2+y^2)/2\sigma^2}$$

3.算法实现

对同一张图片使用 Prewitt, Sobel, Canny, and FDoG 算法

实例 1：

原图

使用 prewitt 算法，阈值 =15

使用 sobel 算法，阈值 =80

使用 Canny 算子，上界=0.20 *最大梯度 下界 = 0.15 * 最大梯度

使用 FDoG 算法

实例 2：

原图

prewitt 算法

sobel 算法

canny 算法

综合上面的测试样例我们可以发现

1. prewitt 算法的细节都比较高（因为设置的阈值较低），概括性稍微下降。
2. Canny 算法的结果是一个二值的图像，边缘最为明显。
3. 从视觉上来说，整体上效果最好的是 FDoG 算法。
4. 图像的最终效果与阈值的选取有很大的关系，一般来说，阈值越大，图像的细节就越少。

4.FDoG

4.1 解释为什么会产生不同宽度的边

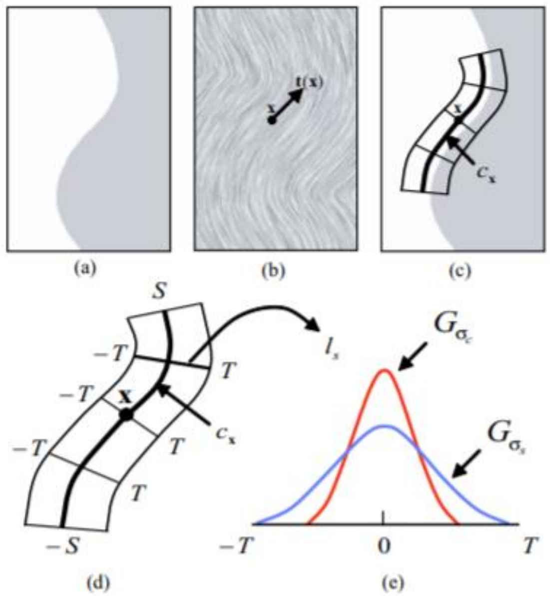
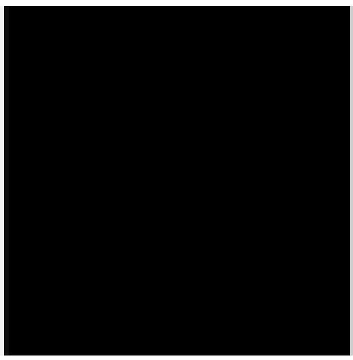


Figure 5: *Flow-based DoG filtering: (a) Input (b) ETF (c) Kernel at* **x** *(d) Kernel enlarged (e) Gaussian components for DoG*

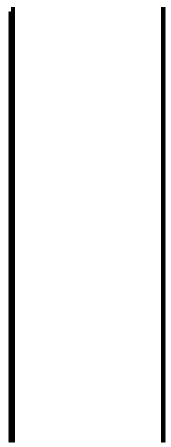
在生成边缘的时候，由于使用了高斯核，每个边缘都会与周围的边缘相关，所以会产生不一样宽度的边缘。

4.2 如何产生一个像素的边缘

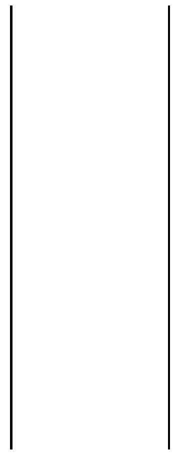
在上图的 e 中，只要降低σ_e的值，就可以降低周边像素的影响（曲线就会越尖），当σ_e小于某个值时，边缘宽度就会降低 1 个像素



原图



较高σ_e



较低σ_e

可以看到，在降低σ_e时，结果从右侧图形（2 个像素）变为左侧图形（1 个像素）

Reference：[1]FDoG 工程地址 https://github.com/SSARCandy/Coherent-Line-Drawing

[2]FDoG paper

http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.108.559&rep=rep1&type=pdf