outline

- 图像点处理方法
- 直方图均衡化
- 图像平滑
- 图像锐化
- 非线性图像处理

图像点处理方法

图像的的处理都是针对每一个像素点，且只与当前的像素点有关，也称之为无记忆性。

可以用数学表达式表示为：

$$g = T(f)$$

其中，f为输入图像，g为输出图像，T为变换函数。

典型的点处理方法：

- Gamma Correction

  $$g = T(f) = cf^{\gamma}$$

  其中c为常数，通常设置为1，γ为参数，主要负责调整图像的**亮度**和**对比度**。- $\gamma > 1$ 图像变暗，高亮度部分被压缩，中间值降低，所以图像变暗

    - $\gamma < 1$ 图像变亮，低亮度区域会得到较大的输出值，提高图像的亮度和暗部细节

- $\gamma = 1$ 图像亮度不变

- Log变换

$$g = T(f) = c\log(1+f)$$

其中，f为输入像素值，通常先归一化到[0,1]之间，c为常数，控制图像变换的**亮度**

- 由于log本身的特性，其导数在暗处比较大，经过log变化后，其变化的幅度会更大，而在亮处比较小。故会将暗部的细节放大，而且防止亮部的信息过多，达到比较好的动态平衡的效果。
- 参数c负责变换后的整体的亮度水平，c值越大，图像越亮。

直方图均衡化

直方图均衡化的主要目的是：输出图像的直方图尽可能的均匀，避免图像聚集在亮度或者暗部区域，是图片包含更多细节，更具有层次感。适合于整体过暗或者过亮的图像，可以展示更多的细节。

输入图像的累积分布函数为：

$$c_f = \sum_{t=0}^{f} p_f(t) = \sum_{t=0}^{f} \frac{n_t}{n}$$

$$g = T(f) = round[\frac{c(f) - c_{min}}{1 - c_{min}} L]$$

其中，L为最大的灰度级，c_{min}为累积分布函数的最小值。

图像平滑

图像平滑的主要目的是去除图片中的噪声和细节，核心是利用低通滤波器抑制图像中的高频成分，保留低频信息。可以在时域和频域操作。

- 时域线性滤波

$$g(x,y) = f(x,y) * h(x,y) = \sum_{j=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} h(i,j) f(x-i, y-j)$$

- 需要注意的是这里的卷积计算时将h翻转180度，然后再进行计算。有的资料卷积计算不会翻转的原因是在设计h的时候就已经翻转了，之后在进行运算。
- 典型的滤波器
  - 均值滤波器
    - 均值滤波器将所有的像素进行平均，可以有效的去除高斯噪声，但是会导致图片模糊
  - 高斯滤波器
    - 高斯滤波的系数符合高斯分布，可以比较平衡的去除噪声，并不会模糊。

- 频域滤波

$$G(u,v) = F(u,v)H(u,v)$$

  - 理想低通滤波器

$$H(u,v) = \{\begin{matrix} 1, & \text{if } D(u,v) \le D_0 \\ 0, & \text{if } D(u,v) > D_0 \end{matrix}$$

$$D(u,v) = \sqrt{u^2 + v^2}$$

    - 可以滤除高频信号，保留低频信号
    - 存在振铃问题 主要因为理想低通滤波器的截止频率处发生突变，根据傅立叶变化的性质，这种不连续性会在时域产生振荡和副瓣，也就是Gibbs现象。
      - 频域的响应逆傅立叶变换后的函数为sinc函数，sinc函数在时域无限延伸，而且随着t增加，幅值虽然衰减，但仍然有明显的副瓣。其在时域卷积，在信号的边缘或者不连续处就会出现震荡，导致信号出现超调或者下冲，进而出现振铃现象。

  - 高斯低通滤波器

$$h(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

$$H(u,v) = e^{-\frac{D(u,v)^2}{2D_o^2}}$$

    - 高斯低通滤波可以对高频成分逐步衰减，而不是直接切断，因此其在时域和频域都有良好的平滑性，也不会产生明显的振铃。
    - 二维高斯滤波可以分解为两个一维高斯滤波，减少计算量。

图像锐化

图像锐化主要是抑制低频成分，保留高频成分，进而保留图片的更多细节。

- 理想高通滤波器
- $$H_{hp}(u,v) = 1 - H_{lp}(u,v)$$

  其中H_{hp}为高通滤波器，H_{lp}为低通滤波器。

- 高斯高通滤波器

$$H(u,v) = 1 - e^{-\frac{D(u,v)^2}{2D_o^2}}$$

- high-boost filter

  与高频滤波只提取高频信息不同，high-boost在保留低频信息的同时增强高频细节。

$$g(x,y) = f(x,y) + (A-1)[f(x,y) - f_{lp}(x,y)]$$

  其中：$f(x,y) - f_{lp}(x,y)$ 表示高频分量（Mask掩膜），即为要增强的细节和边缘。当$A=1$时，相当于高通滤波。

非线性图像处理

线性处理滤波相当于对像素点做加权处理，可能会导致细节丢失，出现模糊的现象，针对强噪声会出现无法消除的问题。因此引入非线性滤波：但因为非线性滤波无法直接使用频域方法分析，因此使用Order Statistic Filters的方法分析。

中值滤波

中值滤波的核心思想是：用中心像素点的邻域像素值的中值作为中心点的灰度值。其数学表达式为：$$f(x,y)=median{f(s,t),(s,t)\in S_{xy}}$$

- 使灰度值差异较大的点更接近于邻域，可以消除突然的过大过小的像素点（面积小于滤波器窗口的一半），较大的一场区域没办法消除
- 可以消除椒盐噪声（脉冲噪声）
  - 椒盐噪声：图像中出现大量的黑白点，类似于椒盐。具有随机、极端值（0或255）以及稀疏的特点

    - 针对椒盐噪声，均值滤波器虽然会一定程度上削弱噪声，但会传播噪声
- 针对边界，均值滤波器会使边界模糊，而中值滤波器会保留边界
- 缺点是：对图像自身的点、线等细节也可能会消除

针对一个信号重复使用中值滤波，最终会收敛到一个不再变化的信号，即为“**根信号**”。

- 因为其保留了图片的主体信息，而且消除了高频噪声。在计算机视觉、人脸识别等很多领域直接处理根信号，而不是原始信号。

其他滤波器

最大值滤波

$$f(x,y) = max\{f(s,t),(s,t) \in S_{xy}\}$$

最小值滤波

$$f(x,y) = min\{f(s,t),(s,t) \in S_{xy}\}$$

中点值滤波

$$f(x,y) = \frac{1}{2}[max\{f(s,t)\} + median\{f(s,t),x\},(s,t) \in S_{xy}]$$

Alpha-trimmed 均值滤波

$$f(x,y) = \frac{1}{mn-d}\sum_{(s,t)\in S_{xy}} f_r(s,t)$$

其中：$f_r(s,t)$是去除极端像素后剩余的像素值，d是去除的像素个数，mn是滤波窗口的像素个数。

核心思想是去除极端噪声，后对剩下的像素值做均值。相对均值滤波可以更好的保留边缘信息。

Iterative Truncated Arithmetic Mean Filter

迭代截断算数均值滤波器，集合均值和中值的优点，同时避免中值的高复杂度。

均值：L2 范数最优解，最小化平方误差

$$\mu = \arg\min_{\phi}\sum_{i=1}^{n}(x_i-\phi)^2$$

中值：L1范数最优解，最小化绝对误差

$$\phi = \arg\min_{\phi} \sum_{i=1}^{n} |x_i - \phi|$$

在存在极端值的情况下，均值和中值的差距会很大。

为了评估均值和中值之间的关系，引入其他三个统计量：

- 上下分位数均值的差值

$$\tau_1 = \frac{1}{2}(\mu_h - \mu_l)$$

其中，μ_h 和 μ_l 分别为数据集中上半部分和下半部分的均值。此数据反映数据的对称性，若数值较大，则偏态严重。

- 标准差

$$\tau_2 = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2}$$

反映整体数据的波动性，如果数值较大，则数据包含极端值

- 平均绝对偏差

$$\tau_3 = \frac{1}{n}\sum_{i=1}^{n} |x_i - \mu|$$

反映数据的变异性，相对标准差，极端数值不敏感一些

其满足以下定理：

- 在任何有限的数据集中满足：

$$|\phi - \mu| \le \tau_1, \quad |\phi - \mu| \le \tau_2, \quad |\phi - \mu| \le \tau_3$$

均值μ与中位数ϕ 的差距小于标准差、平均绝对方差以及上下分位数均值的差值

- τ_3 在所有上界中是最小的。其是中值和均值偏理程度的最佳度量。

$$\tau_3 \le \tau_1, \quad \tau_3 \le \tau_2$$

- 对于任意有限数据集，如果中值偏离均值，则至少存在一个数据点，使其与均值的差值大于τ_3。

$$\exists x_i, x_i \in X, \quad \text{such that} \quad |x_i - \mu| > \tau, \quad \text{if} \quad \phi \neq \mu$$

－ 如果可以识别出极端值并剔除，则均值可以更接近中值

- 如果中值偏离均值，则ITM算法会单调降低截断阈值，并最终收敛至0。

$$\tau(k) < \tau(k-1)$$

$$\lim_{k \to \infty} \tau(k) = 0, \quad \text{if} \quad \phi \neq \mu$$

- 如果中值和均值不同，会逐渐剔除极端值，使均值逼近中值。逐渐消除极端值的影响。

ITM滤波算法：

- 计算当前均值
- 计算截断阈值，剔除极端值

$$x_i = \begin{cases} \mu + \tau, & \text{if } x_i > \mu + \tau \\ \mu - \tau, & \text{if } x_i < \mu - \tau \end{cases}$$

- 检查停止条件

优势在于：避免了中值排序的计算复杂度，而且逐渐逼近于中值。比均值滤波更好的保留边缘，相比于中值滤波减少了计算量。