

1. 习题

1.1 直方图均衡化

假如对一张图已经进行了一次直方图均衡化的操作，得到均衡后的灰度级 S_{k-1} ，再进行一次直方图均衡化得到 S_k 。

由公式 $S_i = T(r_i) = (L - 1) \sum_{j=0}^i p_r(r_j) = \frac{L-1}{MN} \sum_{j=0}^i n_j, i = 0, 1, 2, \dots, L - 1$

其中， $\sum_{j=0}^i p_r(r_j) = \frac{\sum_{j=0}^i n_j}{MN} = CDF(r_j)$

计算第二次直方图均衡化的 S_k ：

$$\begin{aligned} S_k = T(r_k) &= (L - 1) \sum_{j=0}^k p_r(r_j) = \frac{L-1}{MN} \sum_{j=0}^k n_j, k = 0, 1, 2, \dots, L - 1 \\ &= (L - 1) CDF(S_{k-1}) \end{aligned}$$

又因为，经过直方图均衡化的图片的 $PDF(X) = \frac{1}{L-1}$ ，

所以， 经过直方图均衡化图片的 $CDF(X) = \frac{x}{L-1}$

所以 上式 $= (L - 1) CDF(S_{k-1}) = S_{k-1}$

证毕。再次进行直方图均衡化得到的结果跟第一次均衡化的结果一样。

1.2 空间滤波

	177	420	271	263
	75	218	249	107
i.	-131	-324	-107	-133
	-173	-336	-362	-207

ii. 卷积中正数代表图像中该点出现水平边界，下方亮度比上方高，出现负数代表图像中该点也出现水平边界，上方亮度比下方高。

iii. 题中滤波器可以用来做水平边缘检测，如果旋转 90° 可以做竖直的边缘检测

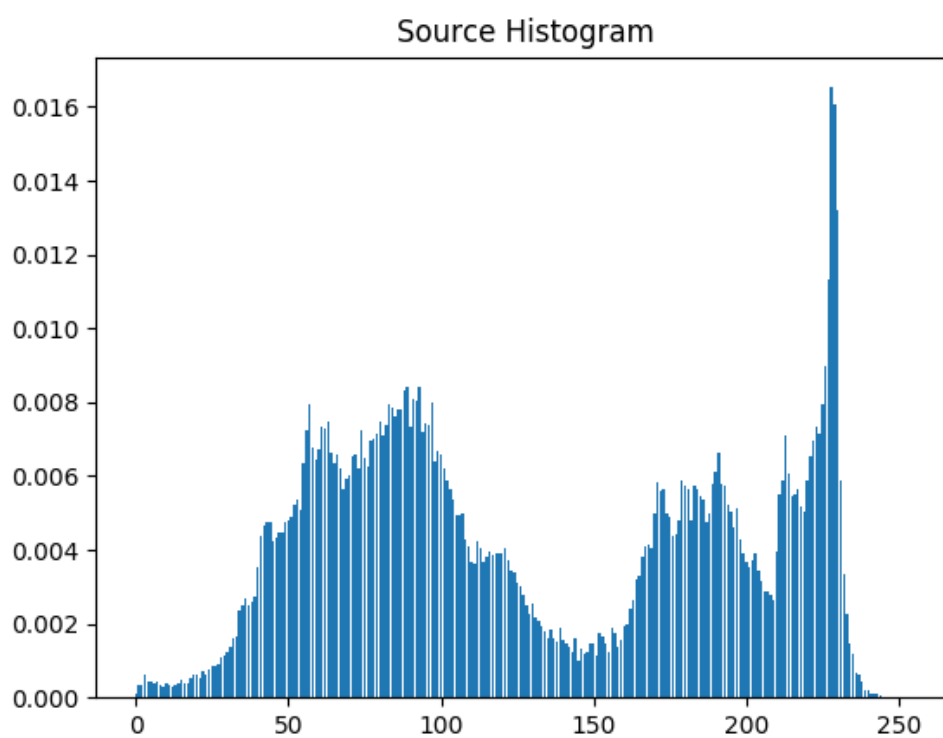
2. 编程题



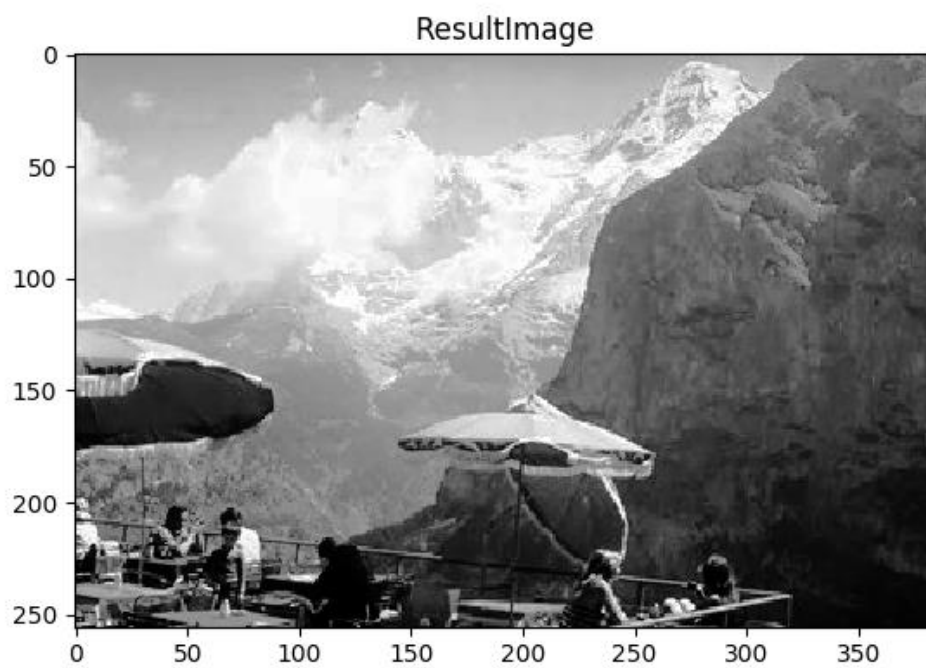
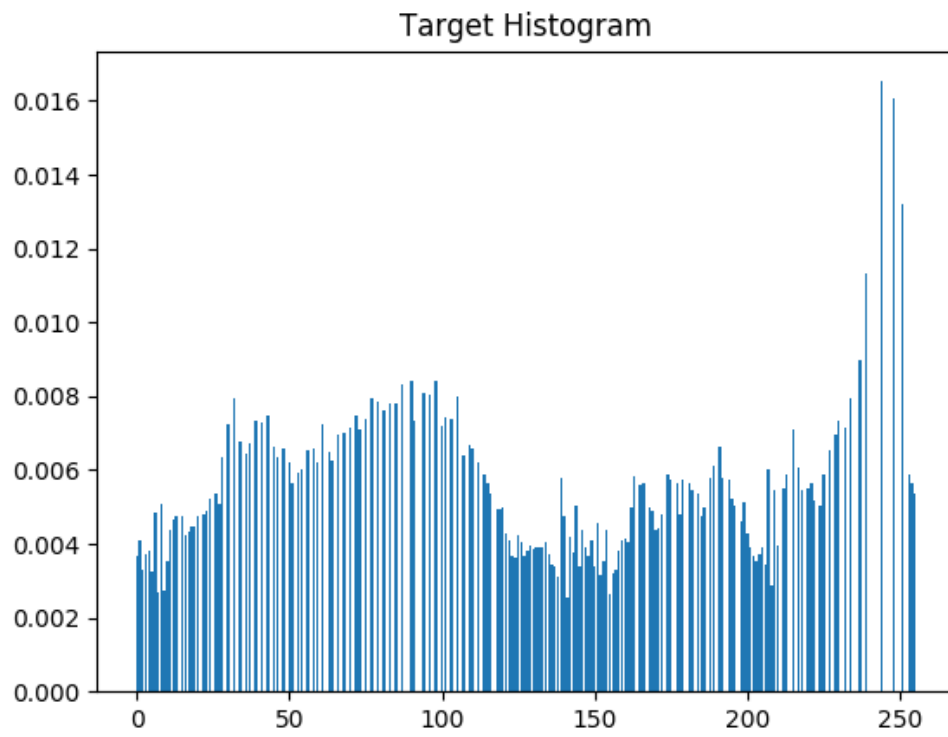
Original Image

2.2 直方图均衡化

1. 计算并显示图像的直方图



2. 进行直方图均衡化，均衡化的结果和相应的直方图如下



3. 分析直方图均衡化后的结果：

对比均衡化前后的直方图，发现均衡化后的直方图分布比较均匀。整体来说，亮暗的对比没有发生太大的变化，而局部的对比（暗处的暗暗对比——亮处的亮亮对比）变得更明显了。

对比均衡化前后的图片，发现均衡化后，图片辨别细节的能力有所加强。

4. 详细描述如何实现直方图均衡化操作：

a) 流程：

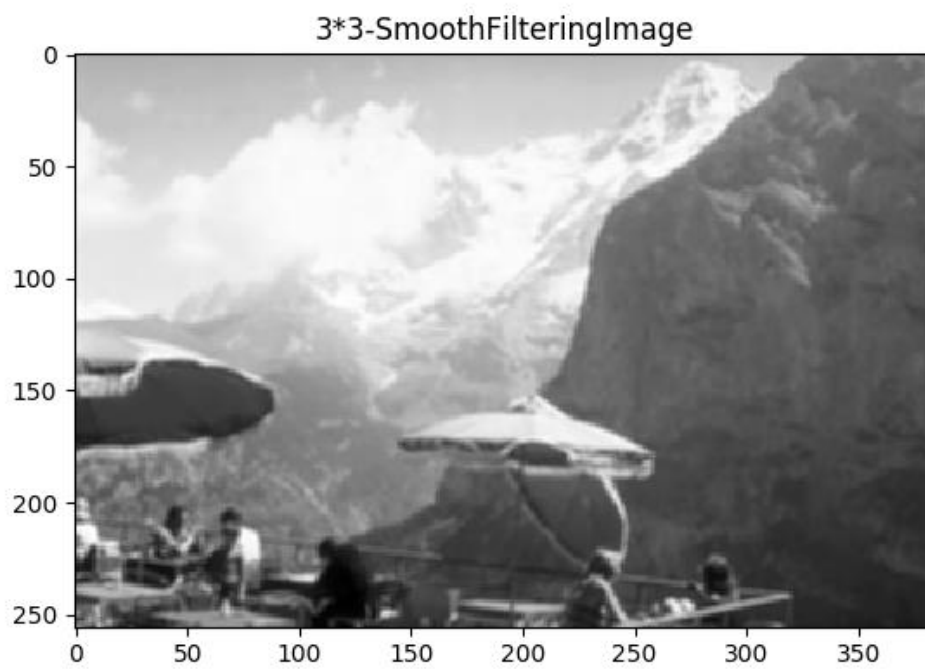
- i. 计算每个灰度出现的频数和对应的频率（概率）
- ii. 通过 I 中频率，获得每个灰度对应的 CDF
- iii. 将每个灰度对应的 CDF 乘以最大灰度值，得到映射后的灰度值，并将所有的映射关系保存起来
- iv. 通过 III 中的映射关系，将图片中的所有灰度值进行映射替换，得到经直方图均衡化的图片

b) 实现

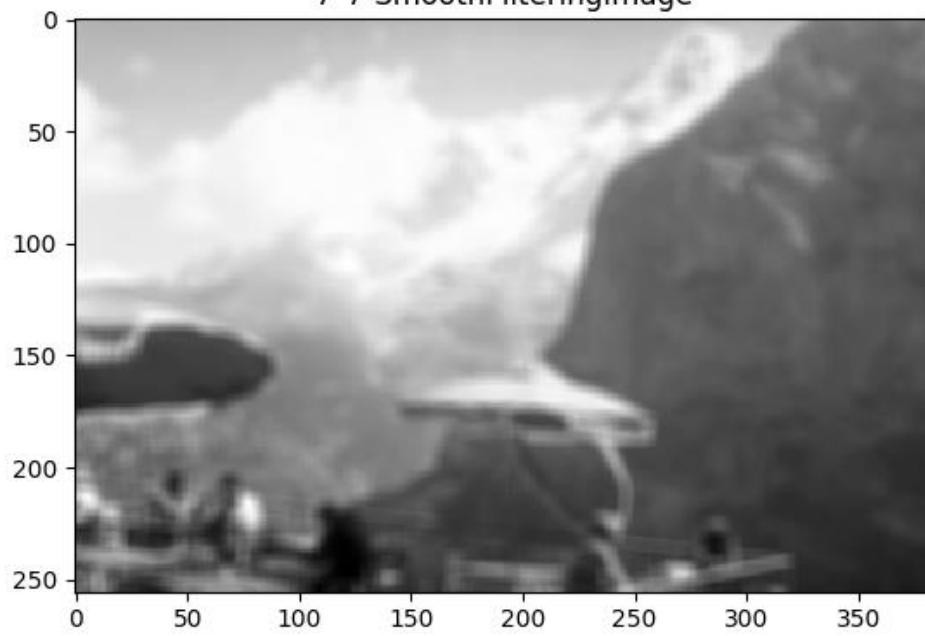
- i. 使用 `list`，其中下标代表灰度值，内容代表该灰度值对应的概率。详见代码 `possibility(img)->list p`
- ii. 得到所有灰度值对应的概率后，通过迭代的方式得到 CDF，乘以最大灰度值得到映射关系，出现小数向下取整，进行灰度替换。详见代码 `equalize_hist(img)->NewImg`

2.3 空间滤波

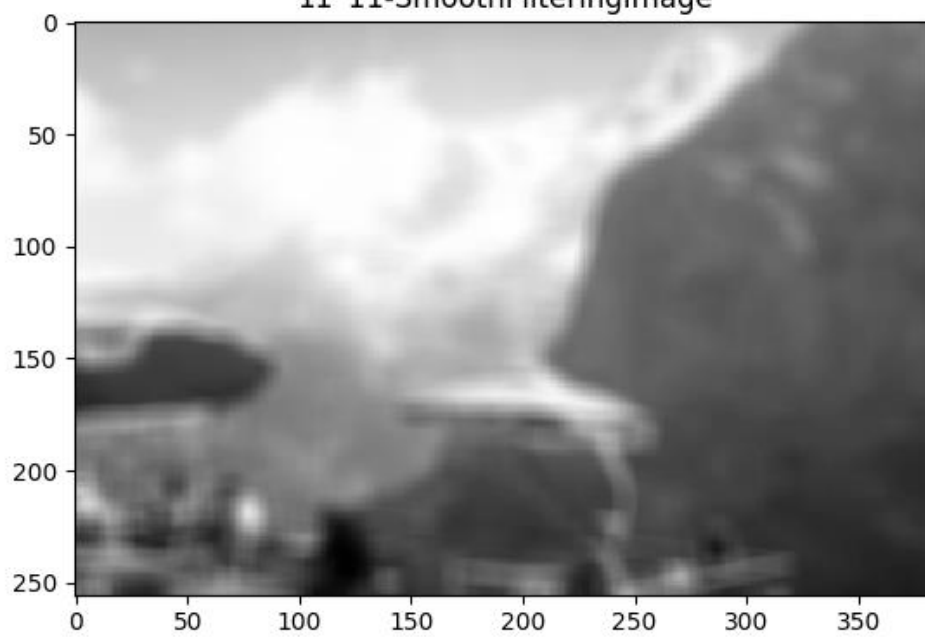
1. 分别用 3×3 , 7×7 , 11×11 的均值滤波器来平滑输入



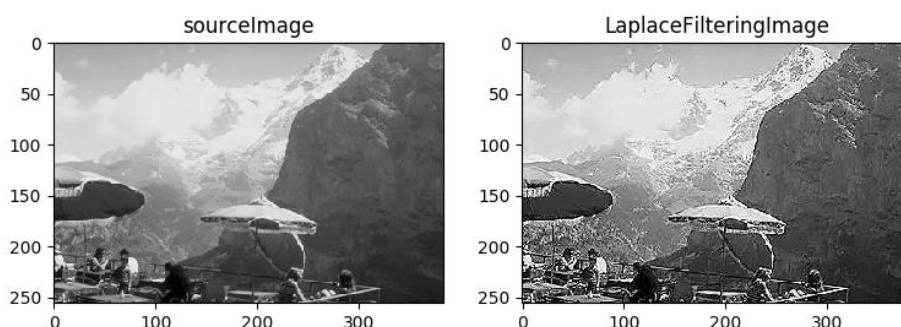
7*7-SmoothFilteringImage



11*11-SmoothFilteringImage



2. 用 3×3 的拉普拉斯滤波器锐化图像



二维图像的二阶微分——拉普拉斯算子:

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

在x方向上，有：

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

在y方向上，有：

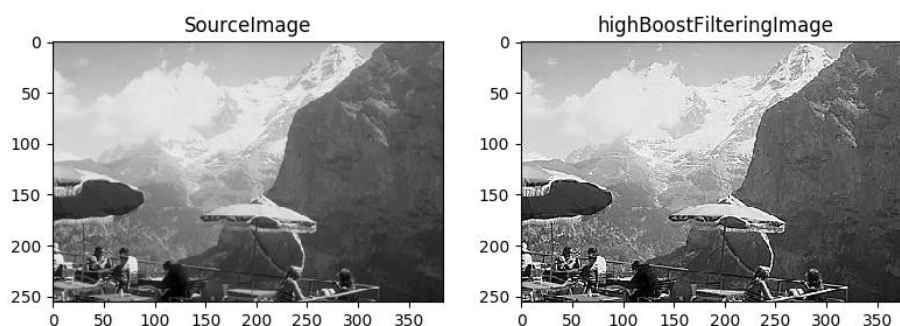
$$\frac{\partial^2 f}{\partial y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y)$$

所以，有：

$$\begin{aligned} \nabla^2 f &= f(x+1, y) + f(x-1, y) \\ &\quad + f(x, y+1) + f(x, y-1) - 4f(x, y) \end{aligned}$$

当算子（滤波器）中心系数为负值时，应将原图像减去该滤波器滤波后的结果，是增强了二阶微分处理后变化剧烈（灰度值突变）的部分。所以，使用拉普拉斯滤波器，可以锐化图像。

3. 高提升滤波用在图像中



选择的 k 的值是 2，使用 3×3 的均值滤波器模糊图像

4. 描述你是如何实现空间滤波操作：

由于本次实验中，是进行滤波操作，且所用到的滤波器均对旋转不敏感，所以进行相关运算。

- i. 图像的边界扩充，图像纵向向扩充 $M-1$ 行，横向扩充 $N-1$ 列（ $M \times N$ 为滤波器的大小）。扩充后，由于担心对边界做相关运算时会出现结果不准确的情况，所以都不采用 0 补充，而是边界拓展。详见代码

`extending(img, filterHeight, filterWidth) -> extendImg`

- ii. 相关运算。将滤波器的各值与图像中相应位置的值相乘求和，并将所得的和放入结果中对应的位置。本次实验中，选定滤波器的左上角为基点，通过 x, y 来做横向和纵向的增量，来得到最终的结果。详见代码

`filter2d(img, filter) -> img` 中的 `calculating` 部分

iii. 裁剪。将 II 中所得的结果进行裁剪，恢复原图像的大小。

详见代码

`Filter2d(img, filter) -> img` 中的 `cutting` 部分

最后，本次实验使用 python3 作为编程语言。使用到了 `numpy`, `PIL`, `matplotlib` 和 `cv2` 等库，最终在 Windows10 的操作系统环境下运行通过。