

1 直方图均衡化在灰度集中的图像中的失真分析

直方图均衡化(Histogram Equalization, HE)是一种常见的图像增强方法,其基本思想是通过灰度级的变换,使得输出图像的灰度分布尽可能接近均匀分布,从而提升图像对比度。设一幅灰度图像的像素灰度值为离散变量 $r \in \{0, 1, 2, \dots, L-1\}$, 其中 L 通常取 256。图像的归一化灰度直方图可表示为 $p_r(r_k) = \frac{n_k}{n}$, 其中 n_k 表示灰度值为 r_k 的像素个数, n 为图像总像素数, $p_r(r_k)$ 表示灰度为 r_k 的像素出现的概率。HE 所采用的映射函数基于输入图像的累积分布函数 (CDF), 定义为

$$s_k = T(r_k) = (L-1) \cdot \sum_{j=0}^k p_r(r_j)$$

其中 s_k 为变换后的灰度值, $T(r_k)$ 为映射函数, 其本质上是对原始概率密度函数的积分。

该变换函数具有单调递增、非线性的特点, 能够将输入图像的灰度分布拉伸至更广泛的范围。然而, 当图像的灰度分布高度集中时, HE 方法会出现严重的问题。例如, 设图像中几乎所有像素的灰度值集中在 $[0, 10]$ 区间, 而其它灰度值几乎没有像素占据。此时, 直方图在低灰度区域聚集, 累积分布函数在 $r_k > 10$ 时变化缓慢甚至趋于饱和, 导致 $T(r_k)$ 在前 10 个灰度等级内迅速上升, 而在其后的灰度区域趋于常数。这意味着原图中微小的灰度差异将被映射为输出图像中的大范围灰度差异, 产生强烈的对比度拉伸。举一个极端的数学例子, 设图像中只有两个灰度值: 灰度为 0 的像素占 90%, 灰度为 1 的像素占 10%。根据映射函数, 可得

$$T(0) = 255 \cdot 0.9 = 229.5, \quad T(1) = 255 \cdot (0.9 + 0.1) = 255$$

也就是说, 原本仅有 1 个灰度单位的差异, 在均衡化后被扩展为 25 个灰度单位以上。原本视觉上几乎不可分辨的黑与深灰, 在处理后被强行拉伸为明亮灰与白, 造成图像严重失真和不自然。

从数学上看, HE 的失真本质在于其映射函数 $T(r)$ 是基于全图灰度的累积分布, 当图像灰度分布不均衡时 (例如集中在暗区或亮区), 变换函数将变得极度陡峭, 导致微小的灰度差异被非线性放大。因此, 虽然 HE 能在一定程度上提升对比度, 但在灰度集中分布的图像中, 其非线性拉伸作用可能导致伪影、过度增强和图像细节的损失, 从而不适用于所有图像场景。

2 实验

我们将使用 Python 语言实现直方图均衡化算法，并对其在灰度集中分布的图像中的失真进行分析。首先，我们生成一个灰度集中分布的图像，并对其直方图均衡化。然后，我们将其与原图进行比较，并分析其失真。

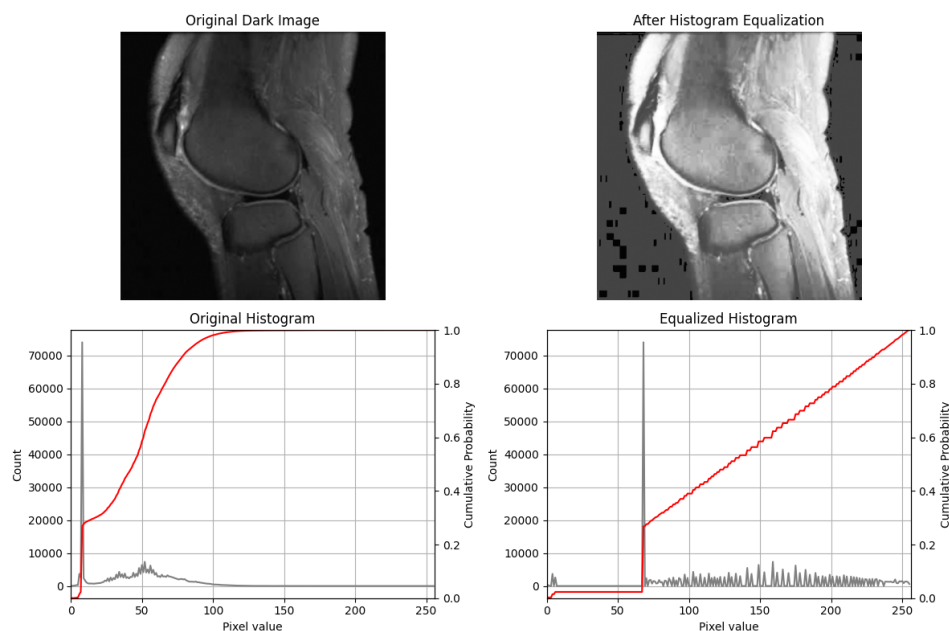


图 1: Example of HE on a grayscale image

显然，由于上图中的黑色区域十分之多，以灰度值在区间 $[6, 8]$ 的为例，大约有80000个，频数最多的灰度值为8累积概率达到了26.82%，利用上一节的算式，可以得出

$$T(8) = 255 \cdot 0.2682 = 68.04$$

在对其他区域进行类似的操作后，原本位于区间的像素点被均值化后变为了 $[68, 72]$ ，使上图中的大部分黑色的区域都被均衡化为灰白色，这就导致了图像的严重失真。

3 CLHE的提出 (Contrast Limited Histogram Equalization)

对于上述问题，一种想法就是对过高的频数的像素点进行截断，使其频数不超过一定阈值，同时再将多余的频数均匀分布到其他灰度值上，以此调控各个频数的数量，从而达到平衡各个灰度值的分布，最终达到减弱失真的效果。

考虑这样一个公式：

$$s_k = (L - 1)T(r_k) = (L - 1) \frac{\sum_{j=0}^k r_j}{n} \quad (1)$$

其中 L 为图像的灰度级数， k 为灰度值 r_k 的频数， n 为图像的总像素数。

为了实现 CLHE，我们需要对输入图像的累积分布函数 (CDF) 进行限制，即限制其在灰度值 r_k 处的累积概率不超过某个阈值 τ ，即

$$r_k = \begin{cases} r_k & \text{if } r_k < \tau \\ \tau & \text{if } r_k \geq \tau \end{cases} \quad (2)$$

这样，当某个灰度值 r_k 的频数超过了阈值 τ 时， r_k 将被截断为 τ ，同时再将多余的频数 $error_k = r_k - \tau$ 均匀分布到其他灰度值上，即对每个灰度值对应的频数增加 $error_k/L$ 。最终我们有

$$s_k = (L - 1)T(r_k) = (L - 1) \frac{\sum_{j=0}^k (r_j + \frac{error_j}{L})}{n} \quad (3)$$

通过该方法，有效地截断了过高频数的像素点，并将多余的频数均匀分布到其他灰度值上，从而达到平衡各个灰度值的分布。

因此，我们提出了一种新的直方图均衡化方法，称为“限定对比度的直方图均衡化” (Contrast Limited Histogram Equalization, CLHE)。

问题没有结束，我们虽然找到了这种截断的方法，但是有一个问题就是如何确定阈值 τ 。笔者的第一个想法是，限制灰度值对应的频数在全体像素中的比例值，即设定一个比例阈值 α ，则对应的频数阈值则为 $\tau = n\alpha$ ，当某个灰度值对应的频数超过了该阈值时，就将其截断为该阈值。