

# Adaptive Unsharp Masking

## 1、作业背景

图像锐化是一种常用的图像增强方式，通过某种计算高频响应得到锐化mask回帖到原图，从而实现边缘等高频信息的增强，提高观感。在实际的锐化使用当中，我们需要提高纹理信息的响应，尽量降低噪声的响应，此时我们需要将图像分为平坦区域、中等强度纹理区域和高强度纹理区域分别进行不同强度的锐化。一般的锐化流程整体可以表示为下图，其中 $x(n,m)$ 表示输入图像，HP filter表示高通滤波器， $z(n,m)$ 表示输出的锐化mask， $y(n,m)$ 表示锐化结果， $\lambda$ 表示锐化强度系数[1]。

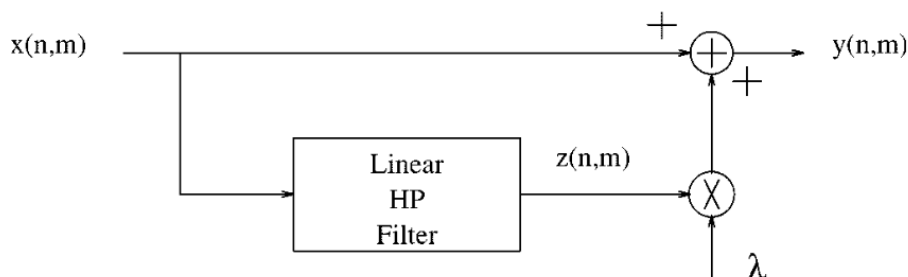


Fig. 1. Unsharp masking for contrast enhancement.

## 2、作业目标

本次作业为开放式作业，没有标准答案，只有参考实现方式。最终目标是通过分区权重锐化来增强弱纹理，适度增强强纹理，少量提升平坦区域噪声。需要思考何种方式可以更好的区分平坦区域、中等强度纹理区域和高强度纹理区域，以及如何设计trick可以更好的提升观感，让图片效果自然不产生视觉断层\*\*。

整体作业大致可以分为以下四个子任务：

**1、实现一种锐化mask（10分）：**锐化方式有多种，在此次题目中，我们将Laplace锐化作为参考锐化mask，可以使用其他锐化mask。

参考：Laplace算子是一个二阶微分算子。有四邻域和八邻域两种，选择任意一种即可：

$$\text{MASK4} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

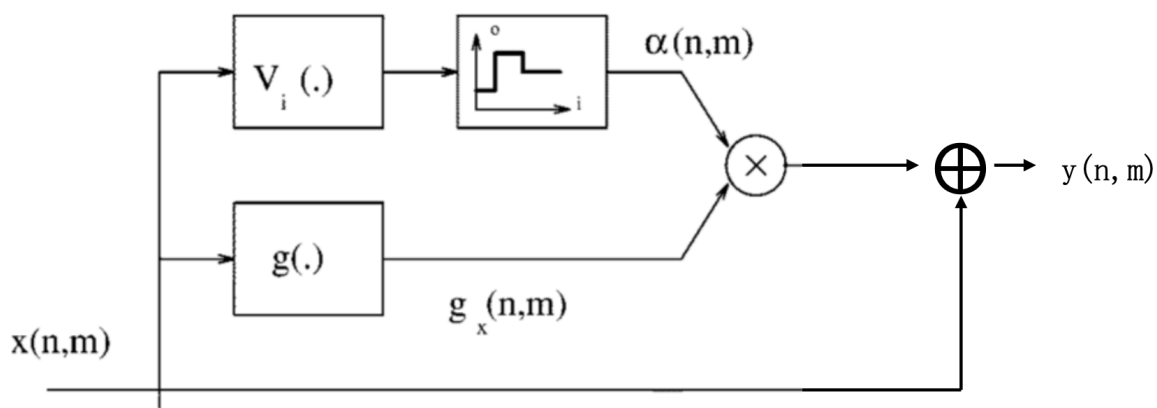
$$\text{MASK8} = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

**2、实现区域的划分（50分）：**开放式任务，目的是将区域划分为平坦区域、中等强度纹理区域和高强度纹理区域。在此次作业中我们给出一个参考方式：计算局部邻域方差，设定两个阈值区分三种区域，可以参考下图公式[1]，两个阈值作为主观调节参数。此题只完成参考方式占25分，其他实现方式占25分(其他实现方式在代码中注释大致思路)。

$$\text{参考： } v_i(n, m) = \frac{1}{9} \sum_{i=n-1}^{n+1} \sum_{j=m-1}^{m+1} (x(i, j) - \bar{x}(n, m))^2 \quad [1]$$

**3、根据划分结果进行不同强度的锐化（10分）：**将目标2划分好的区域设置不同的锐化强度（不一定是如[1]中的分段函数，连续函数可能效果更佳），不同区域根据各自的锐化强度将目标1实现的锐化mask加回到原图上实现适应性分区锐化。不同区域各自的锐化强度作为主观调节参数（无需参考[1]中具体数值设置）。可以参考如下流程， $\alpha(n,m)$ 为锐化强度， $g(n,m)$ 为目标1中实现的锐化mask， $y(n,m)$ 为锐化输出：

参考：



[1]

4、调节参数增加trick提高锐化效果和整体观感（30分）：一个好的锐化效果应当是保持自然的前提下适度增强强纹理，主要增强中等纹理，同时少增强平坦区域的噪声。可以通过调节参数或者加入其他trick使得经过处理后的图片有更好的纹理提升效果，保持自然尽量不产生视觉断层，不出现artifact。

输入图片不同区域主观示意如下：



强纹理



中等纹理



平坦区域

### 3、作业要求

- 1、允许使用opencv的Mat等结构体。
- 2、不能使用opencv filter函数，涉及图像的filter操作需要自己实现。
- 3、尽量减少内存占用，提高运行速度。
- 4、尽量优化代码结构，面对对象编程。

### 4、提交

- 1、需要提交源代码。
- 2、需要提交最终结果的输出效果图，不同方法需在命名中备注。
- 3、提交区域划分结果的可视化输出图（三种区域设置三种不同颜色或者灰度），不同方法需在命名中备注。

### Reference

[1] Polesel A, Ramponi G, Mathews V J. Image enhancement via adaptive unsharp masking[J]. IEEE transactions on image processing, 2000, 9(3): 505-510. Polesel A, Ramponi G, Mathews V J. Image enhancement via adaptive unsharp masking[J]. IEEE transactions on image processing, 2000, 9(3): 505-510.