

## 第四章作业记录

本次主要是实现下面函数：

1. PiecewiseBilateral: 快速双边滤波
2. JointBilateral: 联合双边滤波
3. Gen\_mask: 生成阴影和高光的掩码

### 一、PiecewiseBilateral

理解之后就好实现了，完全对着论文的公式仿写就行。其中插值我没有用 PPT 中推荐的 `interp` 函数，而是按照论文原来的公式去线性插值的，因为总体上运行这一段也不会花太多时间。

注意点

1. 遍历的时候要把最后一段包进去。举例解释，0-255 的图像，按照每段 15 个长度来做，若 `range(0,255,15)`，那么最后遍历到 240 的时候就结束了。这样像素值在 240-255 之间的像素就少了一段：假设  $k$  表示像素值， $J(n)$  表示中心像素当成  $n$  的时候的预测值，那么本来应该是  $k/15 * J(240) + (15 - k)/15 * J(255)$ ，最后少了后面的一段。
2. 涉及到高斯卷积，我一开始是自己计算高斯核，然后调用 `scipy` 相关的二维卷积来做，其中有 `signal.convolve2d`, `signal.convolve`, `ndimage.convolve`，某次执行的时间分别为 8.3s、3.2s、0.15s；但是最后一种有一点误差，因为使用 FFT 来做大核的卷积，这个误差在本次作业中会导致一些点很大，所以就没用最后一种调用。最后的最后，直接调用 `ndimage` 高斯滤波就行了...因为高斯滤波是可以拆分成两个 1D 核来做，所以就更好了。

### 二、JointBilateral

把公式的  $I$  替换成 `flash` 图像即可。此外 `opencv` 中有 `JointBilateral` 滤波调用，可以进行比对，这个帮了我很多。

### 三、GenMask

需要转成灰度图进行比较，我使用的是 YUV 中的 Y。其他的正常按照步骤做，形态学操作顺序时腐蚀、填洞、膨胀。其中填洞在 `opencv` 实现感觉比较麻烦，我是用 `contours` 相关函数写的，即先找轮廓，然后将各个区域填充，没找到什么简便的方法。不像 `matlab` 直接 `imfill(x,hole)` 就行了。

#### 四、各个参数

第3题的细节图像如下:

$$A^{\text{Detail}} = A^{\text{NR}} \frac{F + \epsilon}{F^{\text{Base}} + \epsilon},$$

联合双边滤波的无闪光图

基本双边滤波的闪光图像

你要按下面的公式得到第4题的最终图像:





$$A^{\text{Final}} = (1 - M) \cdot A^{\text{Detail}} + M \cdot A^{\text{Base}}.$$

基本双边滤波的无闪光图像

细节图

1. Abase: 对 noflash 进行滤波, 作为粗糙的结果。空间半径 30, 像素值标准差 0.08, 和论文差不多。
2. Anr: 对 flash 和 noflash 进行联合滤波, 用于细节之处的替换。空间半径 5, 像素标准差 0.05。和论文相比, 半径小了很多 (论文大概是 20-30), 不然很容易模糊; 像素标准差大了很多 (论文大概是 0.001), 感觉论文的参数太容易造成异常点出现了。
3. Fbase: 对 flash 进行滤波, 用于细节之处的替换。空间半径为 3, 像素标准差为 0.12, 同样和论文出入较大, 原因同 Anr。
4. Eps: 即 F/Fbase 那里的值, 为 0.02, 和论文一样
5. M: 即 mask, 检测阴影中需要设置阈值, 表示 flash 减去 noflash 小于多少算阴影, 我设的 0.05

最后结果（重点关注做左边的草堆细节程度、右边背景的噪声水平）：

Abase	
Anr	
Fbase	
Adetail	
AFinal	