**17051634张博程**

**注：作业二的代码运行得很慢，我没有死循环，也没有错误，只要耐心等个五分钟就能跑出来。。。。。**

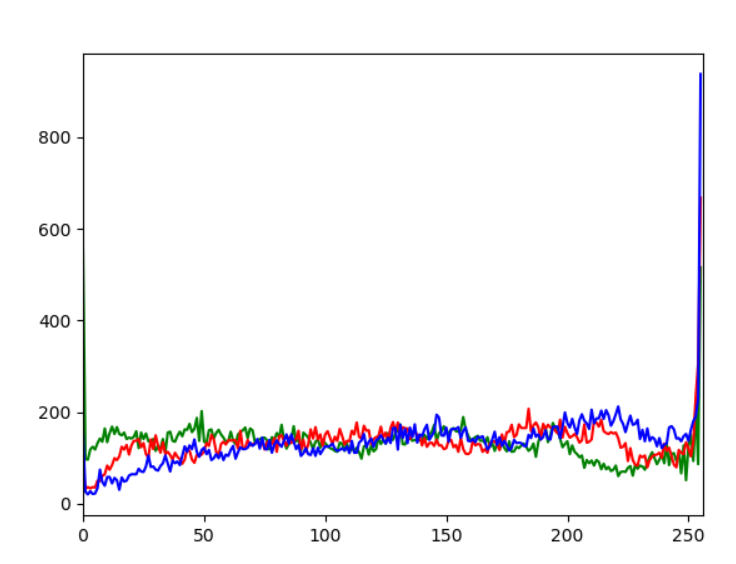
**再注：我的代码都写在一起，运行在最后，我简单地分了些类，有些是测试的，有些是运行的，要运行哪里去掉注释就好。**

**作业一：彩色图像的直方图均衡化**

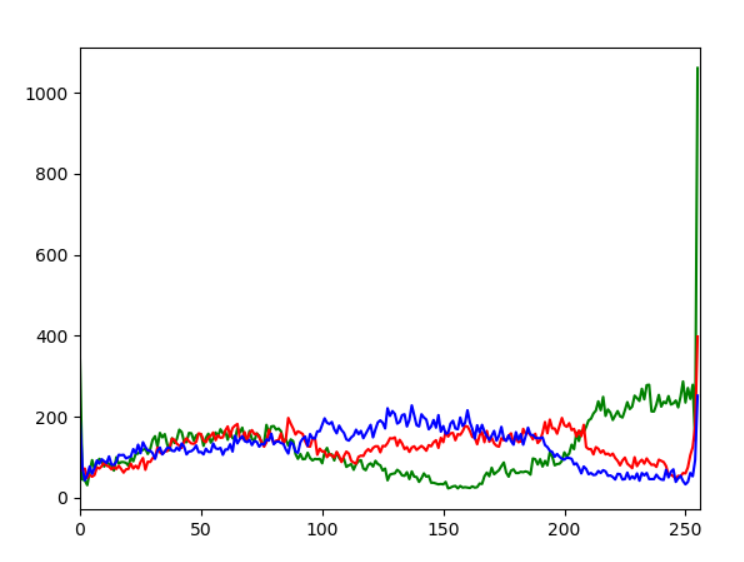
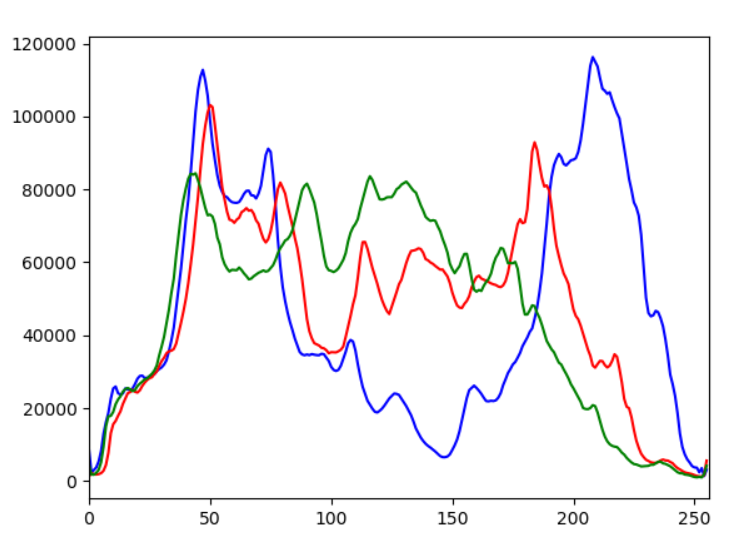
问题1：

对图像中像素个数多的颜色级延展，而对图像中像素个数少的颜色进行压缩，从而扩展取值的动态范围，提高了对比度和灰度色调的变化，使图像更加清晰。

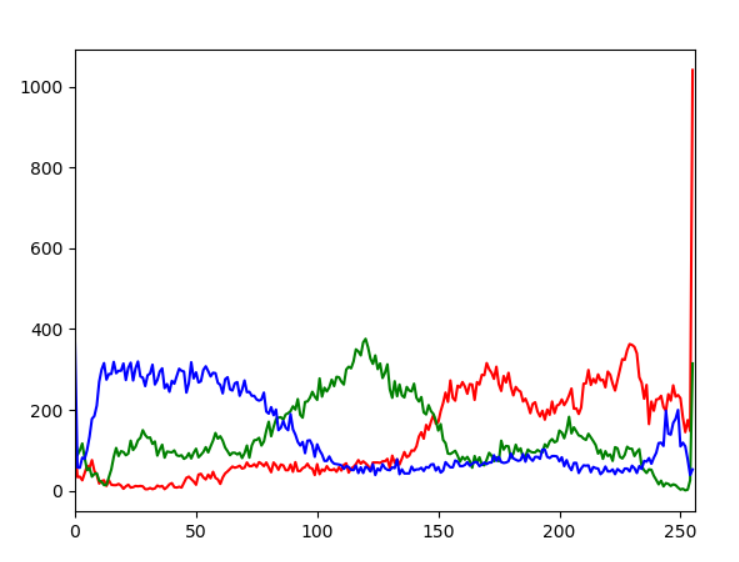
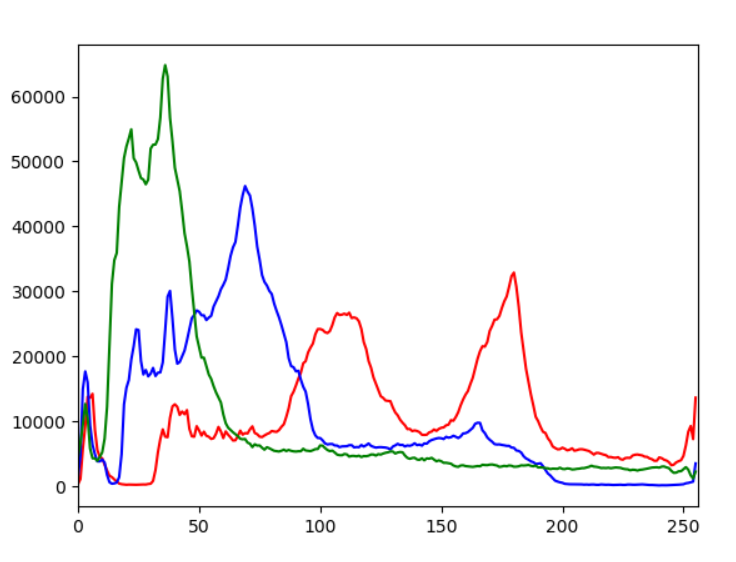


雪地图

凯旋门



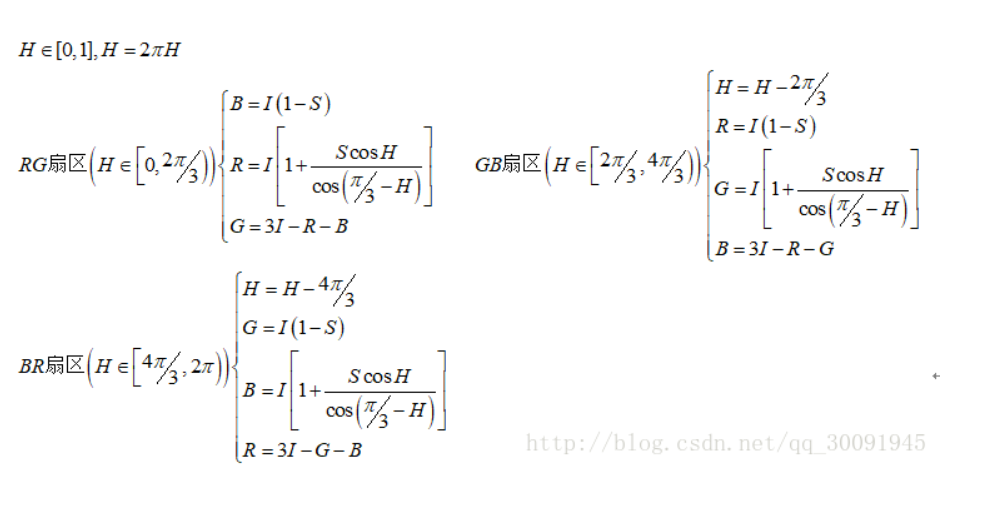
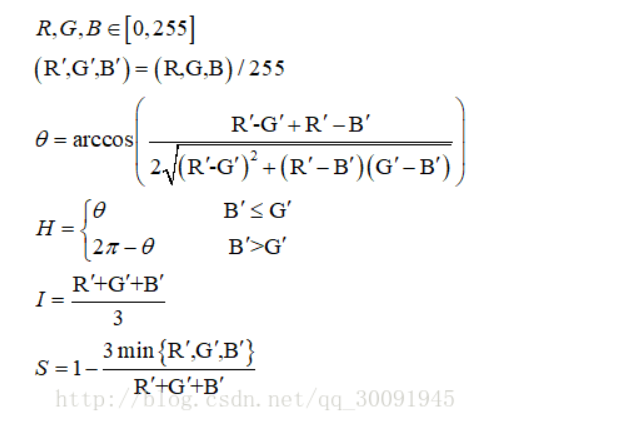
飞机



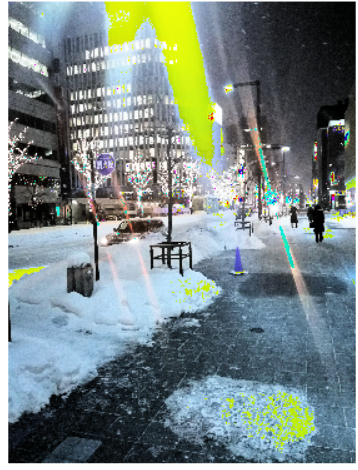
结论：明暗更加清晰了

问题二：

原理：

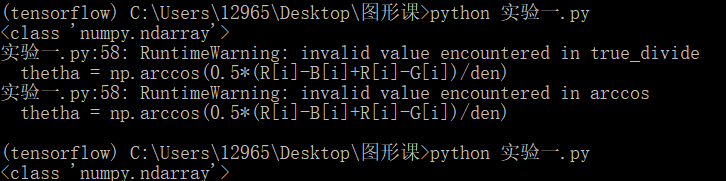


用上面的公式转换成HIS，然后对强度进行均衡化即可，再转回来



（原图和上个作业一样）

结论：强度均衡化后感觉RGB三色都变了，因为和它们都有关。

有个小warning，然后加了个判断就OK了

**作业二：彩色图像的去噪**

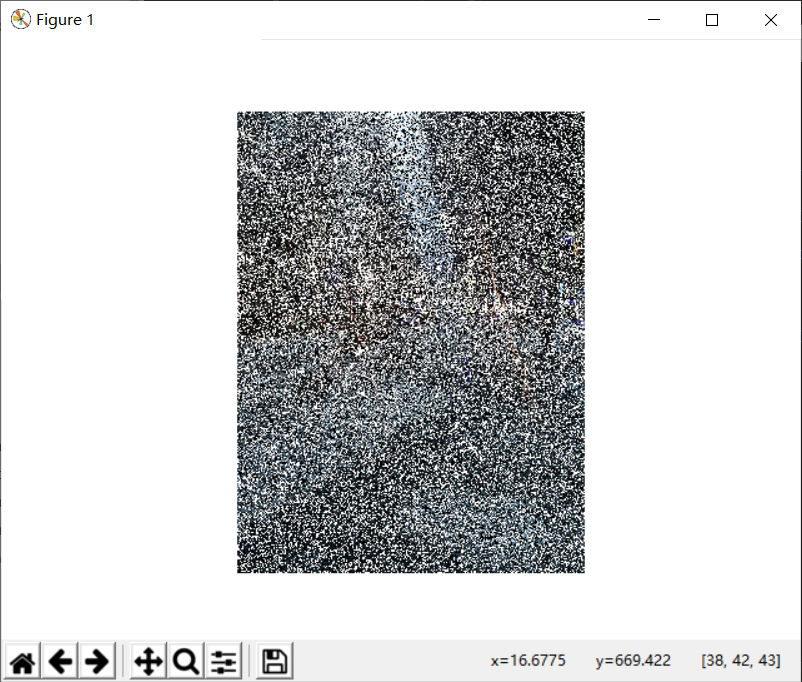
问题一：

椒盐噪声的值为0(黑色)或者255(白色)，这里为我设置的概率的出现0或者255。

高斯噪声：均值决定整个图像的明暗程度，均值默认值为0，大于0表示给图像加上一个使自己变亮的噪声，小于0表示给图像加上一个使得自己变暗的噪声。而对于方差来说，其值越大，数据越分散，噪声也就越多。

椒盐：

对上面那个雪地图加椒盐（为了使效果明显，我设了0.3.。）：



高斯：







结论：那个椒盐噪声加的有些夸张。。其他都还好

问题2：

给噪声乘n\*n的全1的卷积核，由于暴力遍历，我这张高清图要跑五分钟，因为我是分开RGB三个通道跑了在合起来的

就是上面那个椒盐噪声出来的：



对高斯噪声



问题3：

对其n\*n卷积取中值即可，和上个问题很像

对高斯：



对椒盐：



问题2，3结论：虽说是要用中值去椒盐，均值去高斯，但是可能我这个图片问题，我反倒觉得均值对椒盐那个效果不错。。