



# 图像信息处理实验报告

**Digital Image Processing (Experiments)**

|  |  |
| --- | --- |
| 姓 名 | **王祚滨** |
| 指导老师 | **宋明黎** |
| 学 号 | **3180104933** |
| 专业班级 | **信安1801** |

**二〇一九年**

**秋冬学期**

## 实验一

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **实验名称：** | 作业3 | **指导老师：** | 宋明黎 | **成绩：** |  |

### 一、实验目的和要求（必填）

实验目的：

1. 对灰度图进行log提亮操作
2. 对灰度图进行直方图均衡化

要求：

1. 不得抄袭

### 二、实验内容和原理（必填）

实验内容：

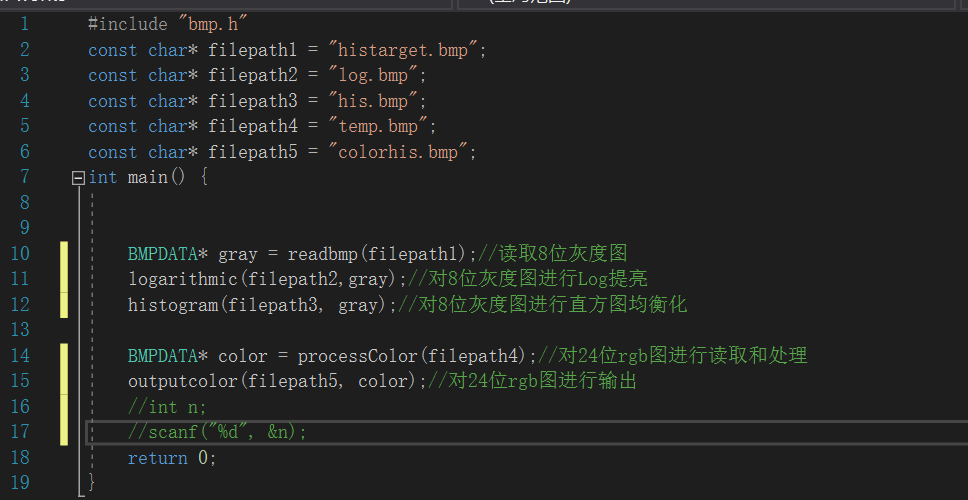
1. 8位灰度图进行Log提亮
2. 8位灰度图进行直方图均衡化
3. 24位rgb图进行直方图均衡化

实验原理：

通过C语言fread,fwrite等函数，配合BMP图片的存储方式，更改像素数据，从而进行各项操作。

### 源代码与分析

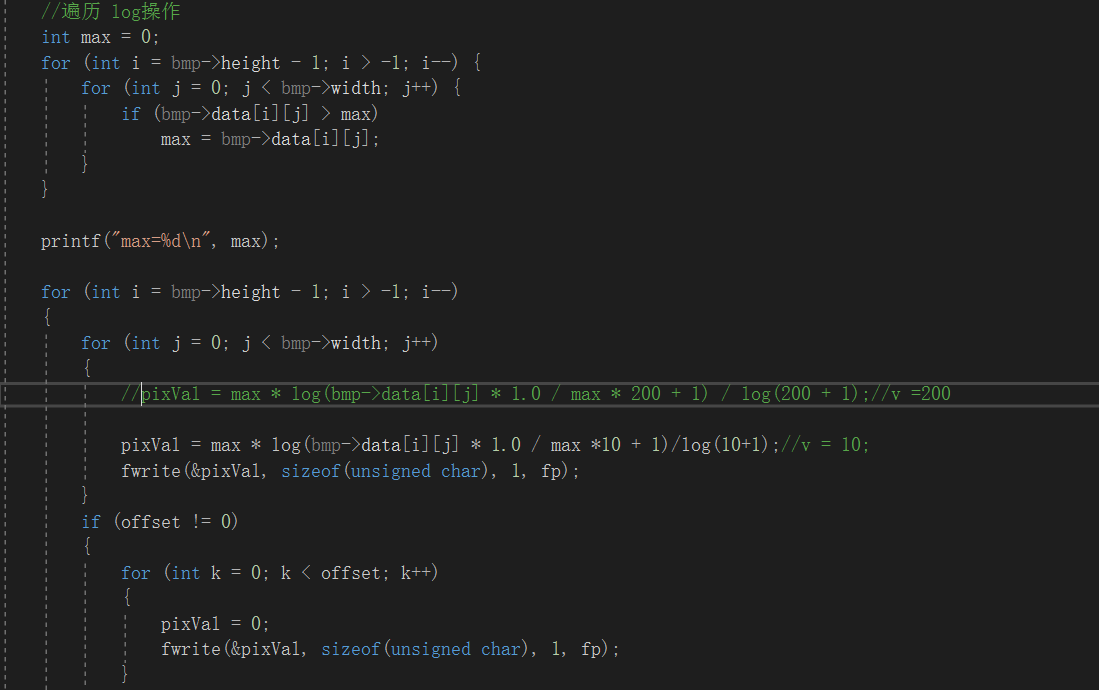
本次实验工程共分为一个头文件与六个函数文件，代码函数分布较为合理。主函数如下：



其中各个函数功能已经注释。

* 1. Log 提亮

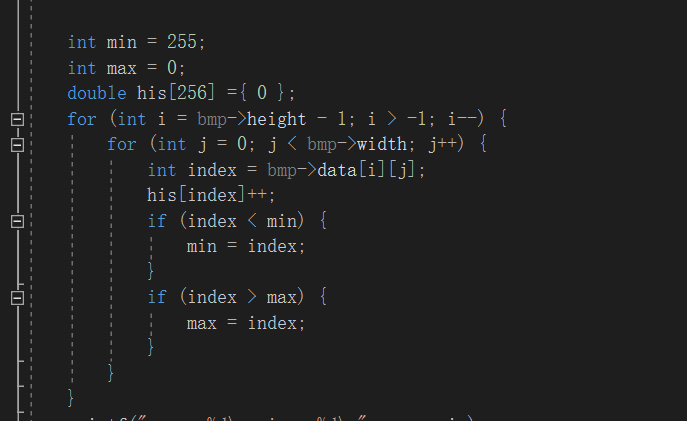
关于代码部分，Log提亮操作最初没有将其映射到[0,1]内，得到的效果有一种灰蒙蒙的感觉，后期更改代码，将其映射到[0,1]内，并更改v值，得到不一样的图像



具体思路为先对每个像素进行遍历，获取最大值，然后再进行一次遍历，其中r=data/max，以此映射到[0,1]范围内。

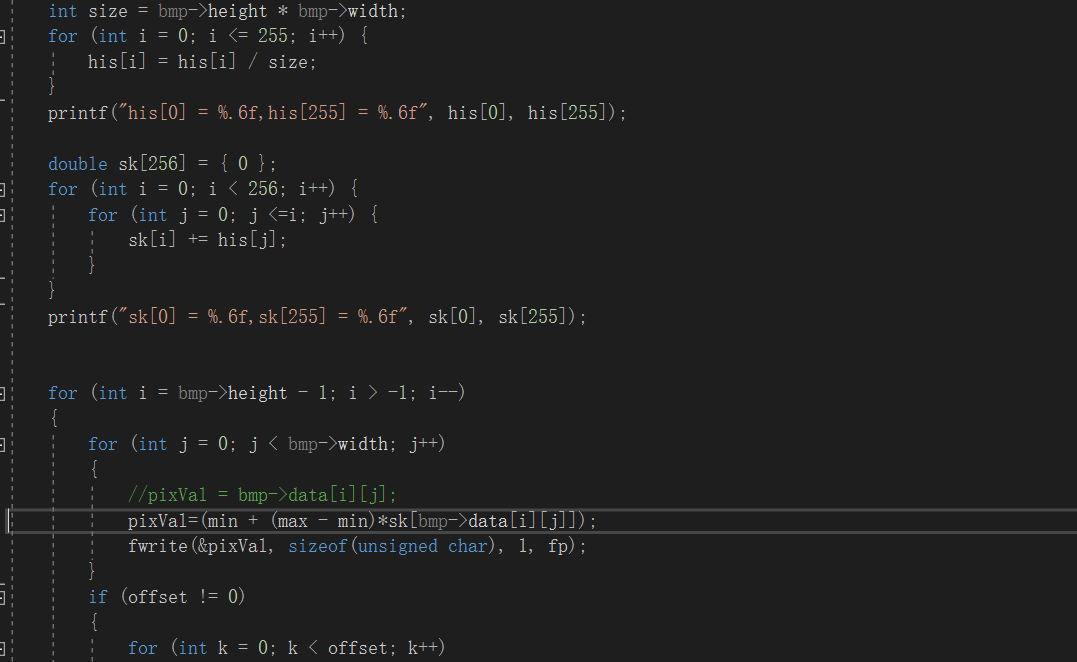
* 1. 灰度图的直方图均衡化

读写操作类似，不过多介绍。



首先遍历像素值，获取到整个图像的max,min值

并且计算每个灰度级元素的个数

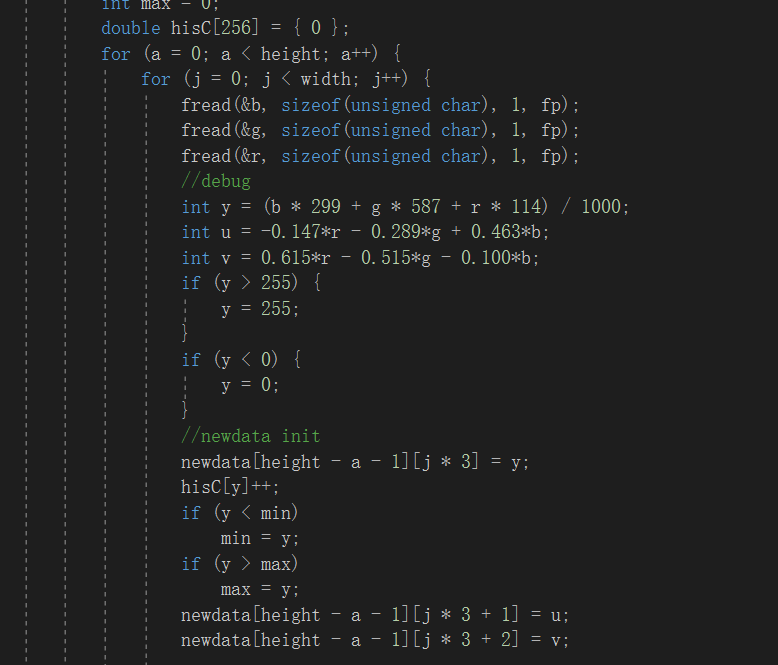


随后计算每个灰度级元素出现的概率，并对其进行累加，保存在sk[256]中；

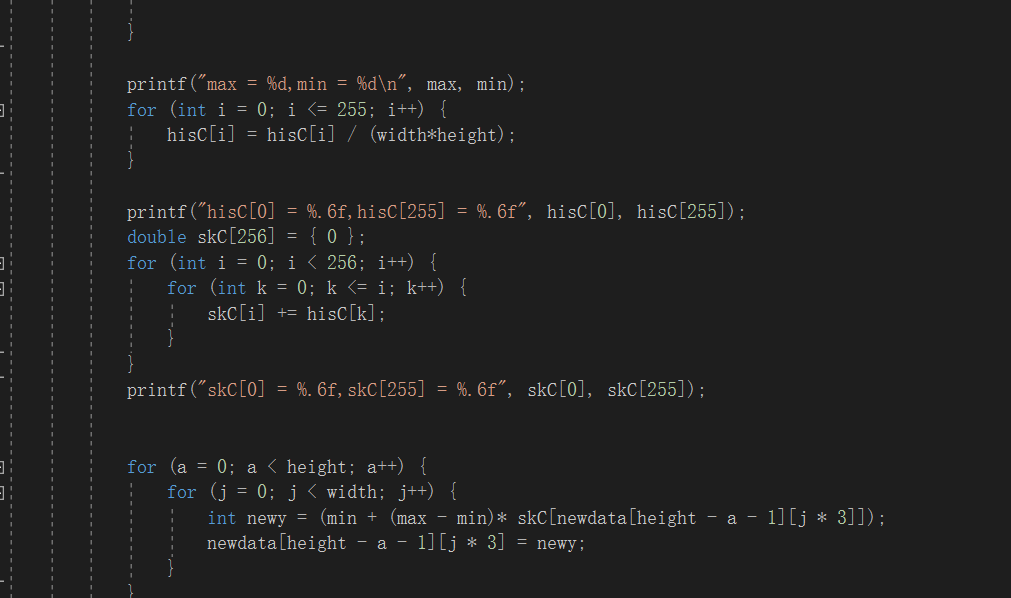
最后利用pixVal=(min + (max - min)\*sk[bmp->data[i][j]]);

对每个像素值进行均衡化，输出图片与原图进行比较。

* 1. 24BIT BMP直方图均衡化



具体思路为先将rgb转换为yuv图像，并针对y通道进行直方图均衡化操作

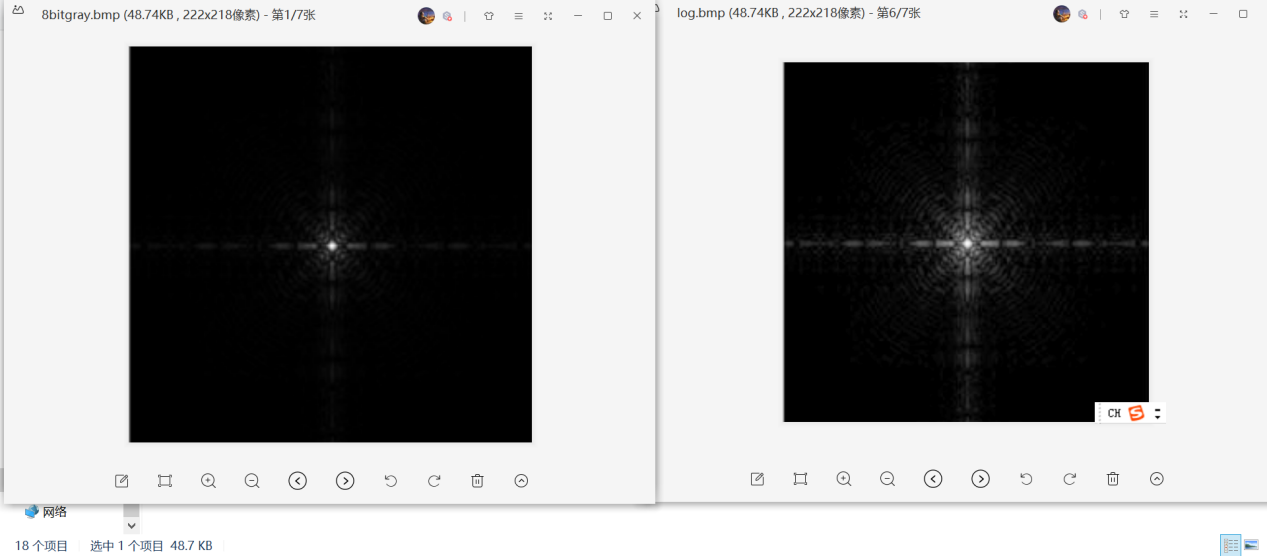


得到newy，再进行还原到rgb图像进行输出。

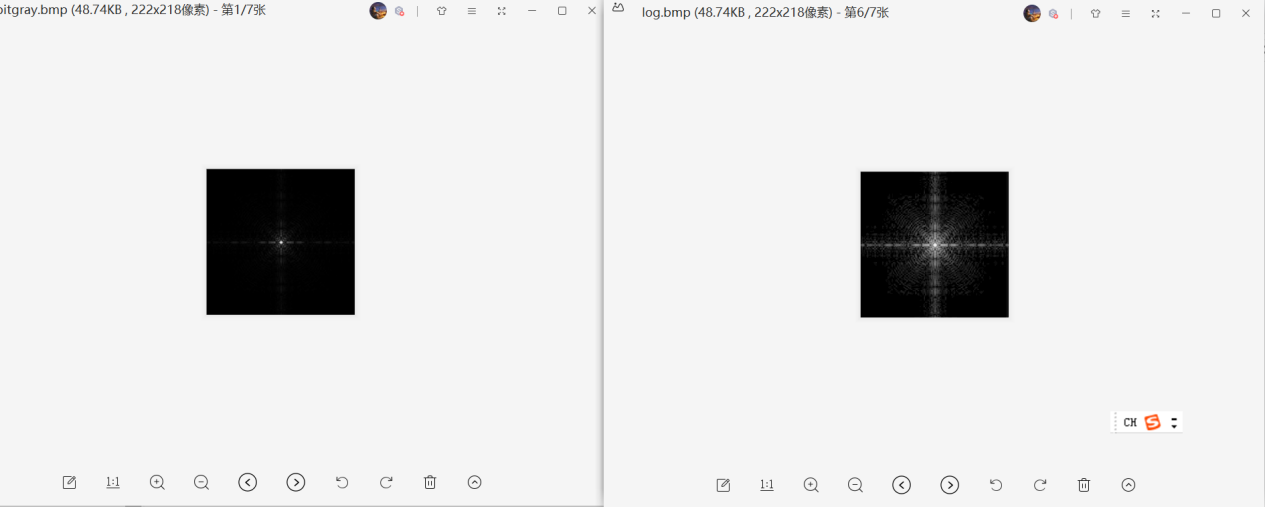
### 问题及心得

本次实验过程中遇到的一些问题总结如下：

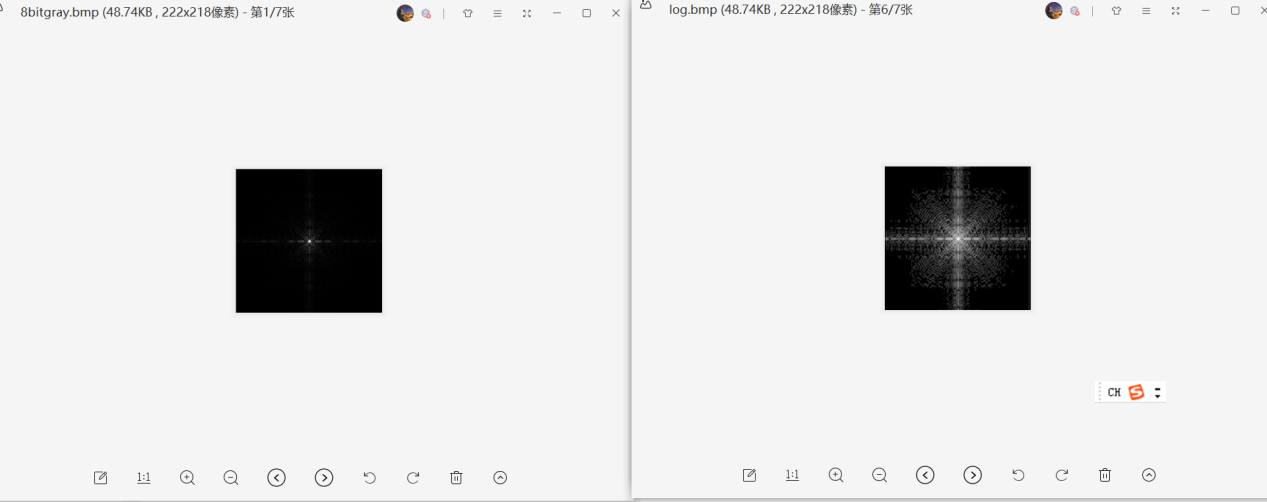
1. 进行Log提亮操作时，首先没有进行映射导致图像有灰蒙蒙的感觉，后经过查阅Log提亮算法，优化代码，成功解决，并测试出不同v值时得到的log图像



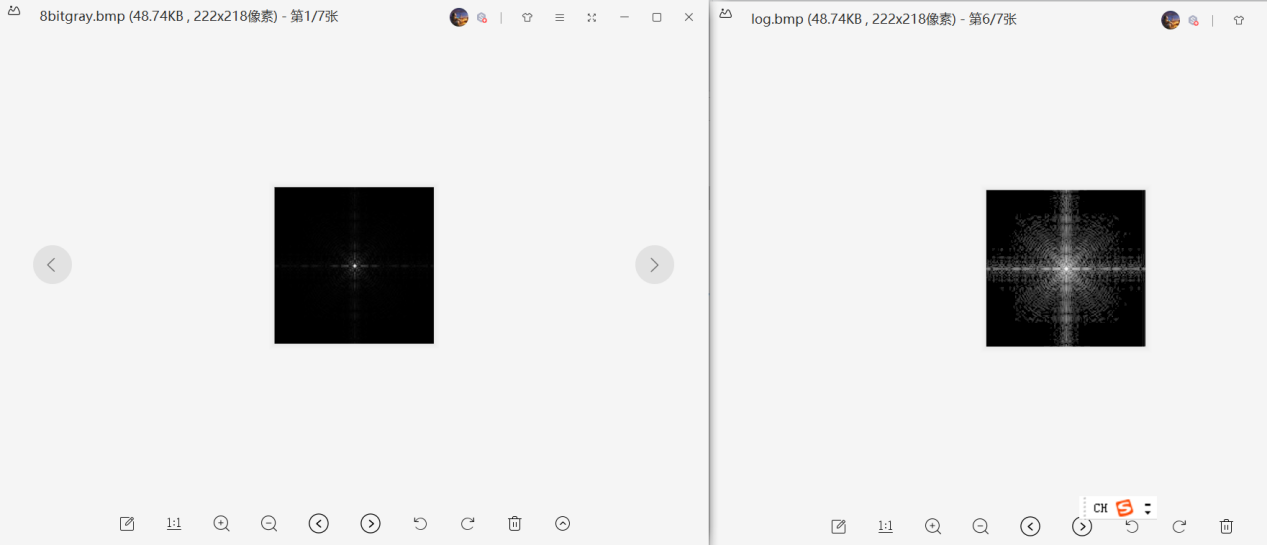
V = 10;



V = 50;



V = 150;



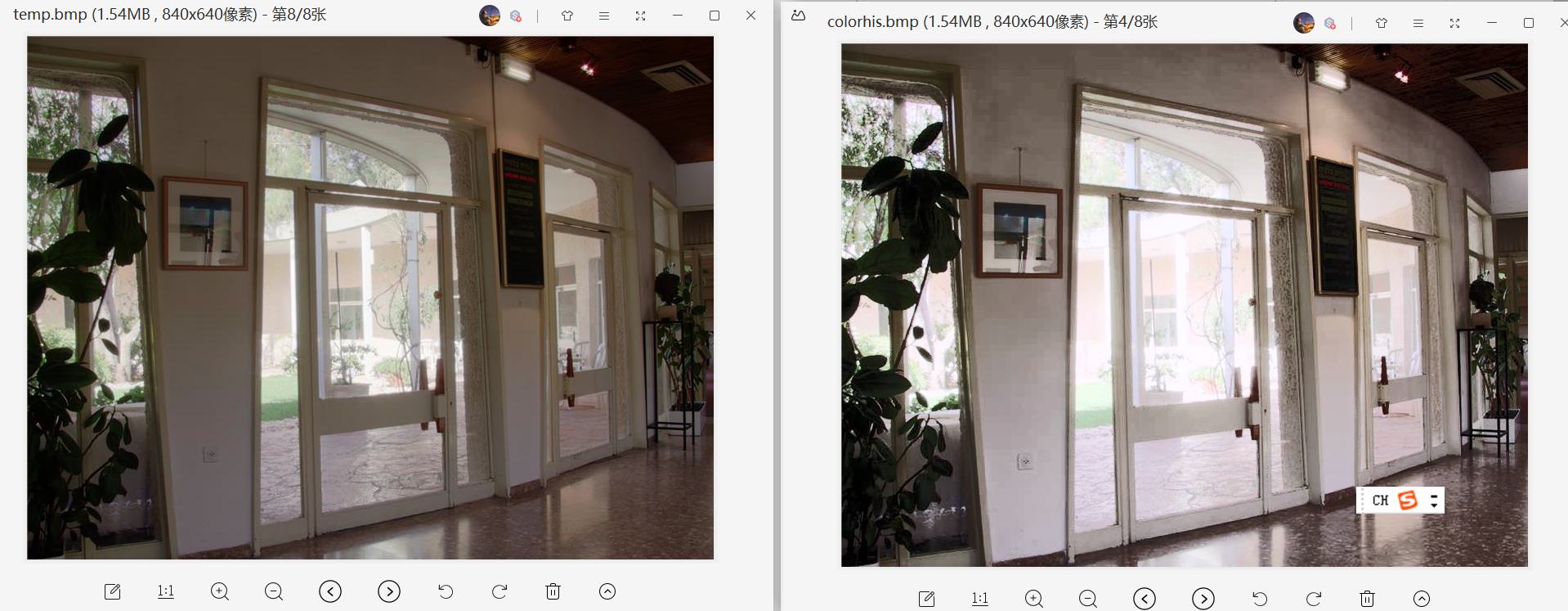
V = 200；

可以看出，V越大，对低灰度部分的强调就越强，对高灰度部分的压缩也就越强。

1. 对RGB图进行直方图均衡化时，得到的图像效果不好

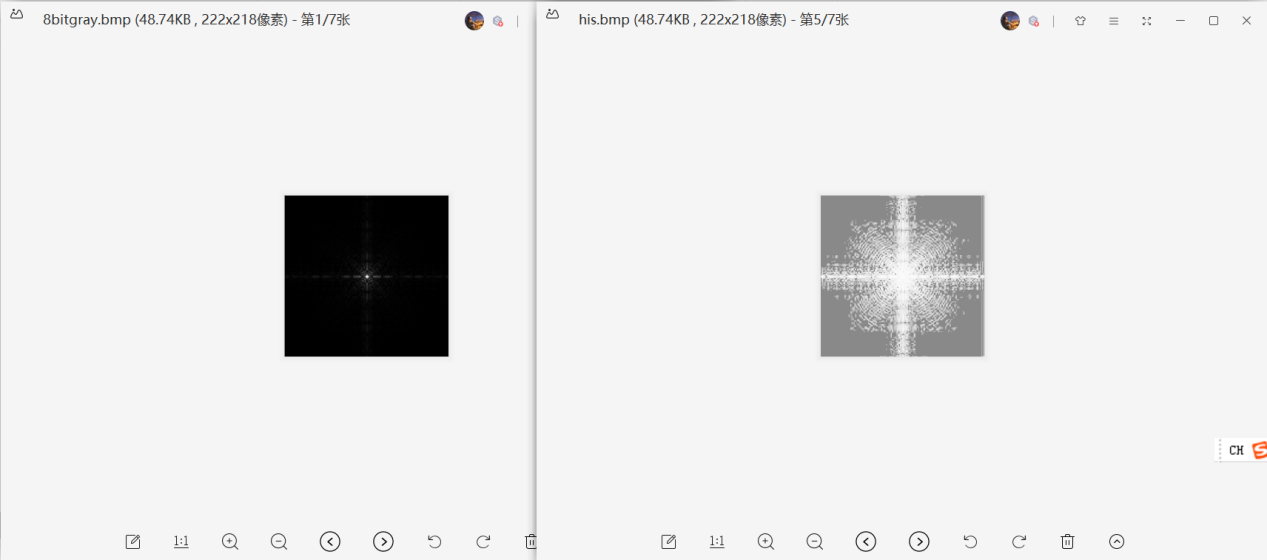
最初采取的方法是对rgb通道独立进行直方图均衡化操作，确实能得到效果，但图像被破坏的程度高，蓝色的天空经过此种方法变成了紫色，得不到想要的效果

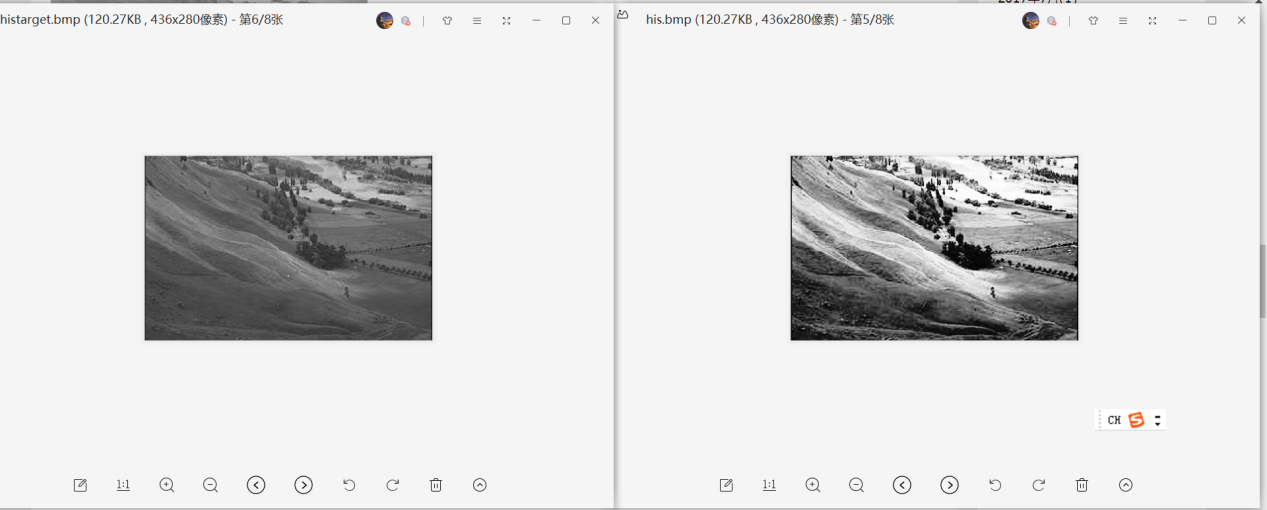
于是改变算法，先将其转换为yuv通道，针对y通道进行均衡化操作，得到的效果明显改善，但老师曾讲，yuv通道实际上并不是相互独立的，LAB通道实现可能会更好，基于时间原因没有尝试，望见谅。



此图为基于yuv通道的均衡化

1. 对于灰度图的均衡化操作





以上为灰度图均衡化操作得到结果。

心得：

通过本次实验，我对BMP文件结构有了更深了解的同时还巩固了老师上课时讲到的一些操作，同时对本次实验的深入思考也让我收获到很多知识，比如V对log提亮的影响，以及YUV通道实际上不独立等等，总的来说此次实验基本达到预期要求。